ALL AMIGA

Amazing COMPUTING™

Your Original AMIGA™ Monthly Resource

Vol. 3 No. 12 DEC. 1988
US $3.95 Canada $4.95

AmiEXPO California 88
•Commodore Announces Two New Amigas

GAMES:
•EMPIRE
•Thexder

Amiga Programming:
•Assembly
•Extending AmigaBASIC
•Easy Menus in JForth

Artificial Intelligence

An Amazing Exclusive: The Creation Of Don Bluth's Dragon's Lair

0 74470 74710 9 12

Our DRIVE is Dynamite!

The AIR DRIVE

Dual-Sided. 880K
Color-coordinated chassis and faceplate
Smaller and lighter than the Amiga drive
External drive pass-through
Extra long 30" interface cable
Spring-loaded dust door
Compatible with all Amiga systems
One year warranty
Internal drive also available

CALL FOR DEALER NEAREST YOU.

$79 Small and powerful. The Avatex 1200E.

Avatex MODEMS

$105	1200P
$99	1200HC
$169	2400
$69	1200i
$159	2400i

FREE WITH EACH MODEM

Amiga communication software
& CompuServe access time.

MEGATRONICS

To Order: 800-232-6342

Ask about our **LOWEST PRICE GUARANTEE**

INSIDE UTAH: (801) 752-2642
FAX: (801) 752-8752
CALL FOR A FREE CATALOG.
CREDIT CARDS VERIFIED FOR YOUR PROTECTION.

MEGATRONICS, INC. BOX 3660, LOGAN, UT 84321

Aunt ARCTIC ADVENTURE

In Aunt Arctic Adventure, you, Charlie the Chimp have talked your good friend Penguin Pete into helping you rescue your Aunt who has been kidnapped by Big Borus and forced to work at his circus in the Arctic. You must battle your way through each of the fifty levels to rescue your Aunt. The game can be played by one or two players.

Mindware International
110 Dunlop Street West
Box 22158
Barrie, Ontario, Canada L4M 5R3

For more information or the dealer nearest you: **1-705-737-5998**
Order line: **1-800-461-5441**

Amazing COMPUTING™

AMAZING FEATURES

AMAZING DEPARTMENTS

I demand a retraction!

Photograph by Laura Audrey

Your Original AMIGA Monthly Resource

• TABLE OF CONTENTS •

Volume 3, Number 12
December 1988

AMAZING COLUMNS

AMAZING REVIEWS

Amazing COMPUTING™

ADMINISTRATION

Publisher:	Joyce Hicks
Assistant Publisher:	Robert J. Hicks
Circulation Manager:	Doris Gamble
Asst. Circulation:	Traci Desmarais
Asst. Circulation:	Donna Viveiros
Corporate Trainer:	Virginia Terry Hicks
Traffic Manager:	Robert Gamble
International Coordinator:	Marie A. Raymond
Marketing Manager:	Ernest P. Viveiros Sr.

EDITORIAL

Managing Editor:	Don Hicks
Editor:	Ernest P.Viveiros Jr.
Hardware Editor:	Ernest P. Viveiros Sr.
Music & Sound Editor:	Richard Rae
Copy Editor:	Michael Creeden
Copy Editor:	Amy Duarte
Art Director:	William Fries
Photographer:	Paul Michael
Illustrator:	Brian Fox
Production Manager:	Donna M. Garant

ADVERTISING SALES

Advertising Manager:	Alicia Tondreau
Marketing Assistant:	Melissa J. Bernier

1-508-678-4200
FAX 1-508-675-6002

SPECIAL THANKS TO:
Buddy Terrell & Byrd Press
Betsy Piper at Tech Plus
Bob at Riverside Art, Ltd.
Swansea One Hour Photo
Boston Jewelry & Loan of Fall River

Amazing Computing™ (**ISSN 0886-9480**) is published monthly by PiM Publications, Inc., Currant Road, P.O. Box 869, Fall River, MA 02722-0869.

Subscriptions in the U.S., 12 issues for **$24.00**; in Canada & Mexico surface, **$36.00**; foreign surface for **$44.00**.

Application to Mail at Second-Class Postage Rates pending at Fall River, MA and additional mailing offices.

POSTMASTER: Send address changes to PiM Publications Inc., P.O. Box 869, Fall River, MA 02722-0869. Printed in the U.S.A.

First Class or Air Mail rates available upon request. PiM Publications, Inc. maintains the right to refuse any advertising.

Pim Publications Inc. is not obligated to return unsolicited materials. All requested returns must be received with a Self Addressed Stamped Mailer.

Send article submissions in both manuscript and disk format to the Co-Editor. Requests for Author's Guides should be directed to the address listed above.

Amazing Dealers

The following are Amazing Dealers, dedicated to supporting the Commodore-Amiga™. They carry Amazing Computing™, your resource for information on the Amiga™. If you are not an Amazing Dealer, but would like to become one, call PiM Publications, Inc.:

1-508-678-4200

Alaska
That Computer Place — Anchorage
Juneau Electronics — Juneau
Alabama
Abax Data Systems — Huntsville
Alabama Book Store — Tuscaloosa
Command Computers — Birmingham
Madison Book & Computer — Madison
Mel's Photo & Computer Shop — Montgomery
The Computer Nut — Birmingham
The Computer Image — Birmingham
Universal Computer System — Mobile
Arkansas
Commodore Connection — Sherwood
Microtronix — Fort Smith
The Micro Shop — Little Rock
Arizona
Books Etc. — Tempe
Books Brothers Ltd. — Tucson
Computer Library, Inc. — Tempe
Computer Works — Glendale
Copperstate Business Systems — Phoenix
North American Digital — Tucson
Try Tech Inc. — Pheonix
California
3 E Software And Systems — Hayward
American Mags & Books Ltd. — Englewood
Augnet — Fremont
Balboa Office Supply — San Diego
Bits & Bytes — Fairfield
Bits n Pieces Comp Prod. — Anaheim
Bookland — Marysville
Brown Knows Computing — Redlands
CAL-Tech Bookstore — Pasadena
Candy Computer — Elk Grove
Centerfold News — Los Angeles
Central Park Bookstore — San Mateo
Century Computer Systems — La Habra
Circus of Books — Los Angeles
Circus of Books — Los Angeles
Commodore Land — Alhambra
Commotech — National City
Computer Attic — Palo Alto
Computer Literacy — Santa Clara
Computer Literacy Books — Sunnyvale
Computer Nook, Inc. — San Bernadino
Computer Outlet — San Diego
Computer Spectrum — Burlingame
Computer Time — Citrus Hts
Cornet Newsstand — Oceanside
Covina Bookstore — Covina
Dataphile — Watsonville
Delaurer's News Agency — Oakland
Discount Computer Station Inc. — Long Beach
Empire Computers — Santa Rosa
Encinitas News — Encinitas
ECK Computer Co. — Walnut Creek
Fisher Computers + Software — Rancho Cucamonga
Harding Way News — Stockton
Heinrichson, Schneider & Young, Inc. — Berkeley
Home Computing Center — Concord
Home Computing Center — Mill Valley
Home Computing Center — San Leandro
HT Electronics — Sunnyvale
International Telecine — Santa Ana
K J Computers — Granada Hills
Little Professor Books — Placentia
LTC Computers — Santa Ana
Lyon Enterprises — Berkeley
Micro Flight — San Jacinto
NewsBeat — Davis
Northridge Newsstand — Northridge
Op Amp Tech Books — Los Angeles
Paras Newsstand — San Diego
Pat's Electronics — North Highlands
R&R Software — Chula Vista
Ridgecrest Computer Ctr. Inc. — Ridgecrest
Sawyer's News Inc. — Santa Rosa
Software First — Santa Rosa
Software Cellar — Anaheim
Software Central — Pasadena
SoftwareLand — San Diego
Software Plus — Citrus Heights
Software Solutions Plus — El Cerito
S.O.S. Computers — Los Angeles
Stacy's #5 — Modesto
Technitronics Service — South Pasadena
The Booksmith — San Francisco
Third Avenue News — Chula Vista
Tower Books — Chico
Tower Books — Citrus Hts.
Tower Books — Concord
Tower Books — Mountain View
Tower Books — Sacramento
Tower Books — San Mateo
The Computer Room — Scotts Valley
Universal News — Hollywood
WCI Computer Central — Eureka
Warner Engineering — San Diego
Winner's Circle — Berkley
World Book & News — Hoolywood
Colorado
Aurora Newsstand — Aurora
Citadel Computers — Colorado Springs
Computer Discount — Denver
Computer Reports — Boulder
Computer Room — Aurora
Micro Computer World — Fort Collins
Micro World — Lakewood
Sun Country Computers — Colorado Springs
Connecticut
Aetna Life Club Store — Hartford
Amiga Business Computer — Norwalk
Computer Outlet — PleVernon
Computer & Software Connections — Plantsville
Deric Electronics — Hamden
Friendly Computers — Plantsville
Mnemonics Inc. — Norwalk
One Byte — Quaker Hill
Personal Computer Center — Norwich
Software City — Orange
Software Kingdom — East Hartford
Tricom Computers — North Windham
District of Columbia
Programs — Washington
Delaware
Castle Video — Newark
Delaware Electronics — New Castle
Newark Newsstand — Newark
Florida
A A Computers — Jacksonville
Chuck's Newsstand — Vera Beach
City Newsstand — Melbourne
Commodore Country Inc. — Pinellas Park
Computer Bar — Pennsacola
Computer Base — Brandon
Computer Image — Miami
Computer Terminal Inc. — Sebring
Computers + Video Creations — Palm Beach
Computers 4 Rent — W. Palm Beach
Computers Plus — S. Daytona
Computer Room — Ft. Lauderdale
Cotronics Inc. — Stewart
Creative Equipment — Miami
Education Computers Etc. — Talahasse
Family Computers — Sarasota
Florida Book Store — Gainesville
Gulf Coast Computer Exchange — PanamaCity
Heartland Computers — Daytona Beach
Joe's News, Inc. — MiamMicro Computer
Center — Largo
Newton's Video — Boca Raton
PC Collection — Coral Gabels
Program Store — Clearwater
Prospect News & Books — Ft. Lauderdale
Random Access Computers Inc. — Ft. Walton Beach
RGB Computers and Video Creations — Palm Beach Gardens
Sunny's At Sunrise, Inc. — Sunrise
Computer Room — Ft. Lauderdale
The Open Door — Merritt Island
Georgia
BFJ Enterprises — Roswell
Cardinal Computers — Dalton
Software South Inc. — Sevannah
The 64 Store — Atlanta
Hawaii
Astro Computers & Electronics — Aiea
Computronics — Honolulu
Data-One Computer Center — Honolulu
J & C Repair — Aiea
Idaho
A.B.I. Computer & Video — Nampa
Boise Computer & Video Inc. — Boise
Illinois
Audio Visual Req. — Chicago
Computer Software Connection — Moline
DigitalWorld — Addison
DOS Computer Center — Niles
I.C.V. Data Center Inc. — Mundelein
Keep In Pace Computers — Champaign
MJJJ Inc. — Champaign
Kroch's & Brentano's — Chicago
Kroch's & Brentano's — Schaumburg
Kroch's & Brentano's — Orland Park
Kroch's & Brentano's — Aurora
Micro Age — Oaklawn
Micro Ed Enterprises Inc. — Chicago
Micro Tek — Kankakee
Ring Software — Geneva
Software City — Bellville
Software Plus — Wheeling
Software Plus Chicago — Chicago
Software Plus West — Hanover Park
The Memory Expansion — Bloomington
Unique Computer — Sterling
Warren Radio/FRANCO — Peoria
Indiana
Bookland — Indianoplis
Burkat Computer Ctr. — South Bend
Bytrex of Angola — Angola
Computer Corner — Marion
Computer People Inc. — Michigan City
Computer Products Unlimited — Ft. Wayne
CPU Inc. — Indianapolis
Indiana News — Indianapolis
Micro Computer Centers Inc. — Indianapolis
Reese Equipment Co. — Plymouth
The Book Corner — Bloomington
The Computer Room of Merrivile — Merrivile
Vons Computers — West Lafayette
Iowa
Gronst Computers — Des Moine
Micro Computer Applications LTD — Marshalltown
Soft Search — Ames
Kansas
Arc Electronic Distributors — Wichita
J & L Electronics — Liberal
MicKansas Computers — Newton
Midwest Computers — Manhattan
Software City — Overland Park
Kentucky
Computer Dimensions Inc. — Lexington
Computer Mania Inc. — Florence
Computer Store of Kentucky — Louisville
Hawley Cooke — Louisville
Reality World Computers — Lexington
Louisiana
City Newsstand — Baton Rouge
Computer Time — Shreveport
Lakeside News — Metairie
Modern Business Machines — Metairie
Software Center International — Metairie
Software Source — Lafayette
The Book Shelf — Slidell
The Computer Clinic — Lafayette
Maine
Computer Barn — Damariscotta
Hands On Computing — Westbrook
Radio Shack Associate Stores — Jay
Skylight Software — Belfast
Maryland
Buried Treasure — Rockville
Cal Com Inc. — Silver Spring
Compuvision Vision Center — Columbia
Greetings and Readings — Towson
Knowledgeware Systems — Kensington
Micro Computer Center — Baltimore
Professional Micro Services — Baltimore
Software Advantage — Rockville
Software N Things — College Park
Waldorf Computer — Waldorf
Massachusetts
Computer Zone — North Attleboro
HCS Computer Center — Marshfield
LCA Video & Computer Ctr. — Norwood
Memory Location — Wellsley
OmniTek Computers — Tewksbury
Pioneer Valley Data Equipment — Amherst
Soft Designs — Watertown
Softpro — Burlington
The Bit Bucket — West Newton
The Software Haus — Quincy
Tycom inc. — Pittsfield
Michigan
CompuSoft — Kalamazoo
Computer Emporium — Lansing
Direct Access — Lansing
Central Park Computers — Okemos
Global Computer Center — Dearborn
I/O Software — Flint
Michigan News Agency — Kalamazoo
Michigan Software — Novi
Micro World — Livonia
Micro World — Milford
Micro World — Roseville
Micro World — Southfield
Micro World — Southgate
Micro World — Sterling Heights
Microworld Computer Center — Dearborn
Midland Computer Shop — Midland
Norman Business Control Sys. — Wyoming
Readmore Newscenter — Lansing
Rochester Book Center — Rochester Hills
Roseville Computer Store — Warren
Slip Disk — Madison Heights
Software City — Grand Rapids
Specialists Computer — Livonia
State Street Computers — Ann Arbor
T.J. Enterprises — Clinton
The Edge Connector — Clio
United Computer — Canton
Ye Olde Computer Shoppe — Ypsilanti
Minnesota
Computer Excercise World — Minneapolis
Computer Stuff — Minneapolis
Computers, etc. — Eagan
JMH Software — Maple Grove
Lookahead Software Compute — Richfield
Shinder's — St. Paul
Shinder's — Maplewood
Shinder's — Burnsville
Shinder's — Minnetonka
Shinder's — Edina
Shinder's — Crystal
Shinder's Read More — Minneapolis
Specialists IN — Hopkins
Valiant Inc. — Stillwater
Mississipi
Computer Connection — South Haven
National Computer Center — Ocean Springs
Missouri
Associated Computer Services — Springfield
Brands Mart Computers — Kansas City
Carthage Electronics — Carthage
Data Byte Computer Co. — St. Louis
Software City — Independence
Systems Plus — Overland Park
Systems Plus of St. Louis — St. Louis
Montana
Appelgren Computer Systems — Great Falls
Nebraska
Double E Electronics — Omaha
My Com Inc. — Plattsmouth
Nevada
Century 23 Inc. — Las Vegas
Computer World — Las Vegas
Tower Records — Las Vegas
New Hampshire
Dartmouth Bookstore — Hanover
Diversified Computers — Keene
New Jersey
Business System Professional — Scotch Plains
Family Computer Centres — South Orange
Morris County Stationers — Chester
Software Station — Rockaway
Take One More Byte — Hoboken
The Program Store — Eaton Town
The Program Store — Wayne
Unlimited Software Inc. — Rutherford
New Mexico
New Horizon Computer System — Alamogordo
Page One Newstand — Albuquerque
Read Runner — Espanola
Technological Concepts — Las Cruces
New York
Amuse — New York
Amicus- Hudson Valley — Wappingers Falls
Amiga Business Computer — East North Port
Byte Shop — Merrick
Compurite Center — White Plains
Computer Outlet — Jamestown
Computers Etc. — Syracuse
Computerware — East Meadow
Contemporary Computing — Latham
Creative Computers — Johnson City
Cromland Inc. — Brooklyn
Leeder's Video — Schenectady
Leigh's Computers — New York
MCP Associates — Astoria
Mnemonics — North Port
Readers Market — Tonawanda
Software & Such — Scotia
Software City — Forest Hills
Software Link — White Plains
Software Supermarket — Tonawanda
Star Tech Systems — Massena
Syrfaire Trading — New York
Sysput Computers — Brooklyn
Tadiran — New York
Tedrow Business Products — Rochester
The Computer Cellar — Albany
Unicom Electronics — Johnson City
Video Computer Bin — Brooklyn
Video Computer Center — Rome
Village Computer Inc. — New York
World Computers Inc. — Hicksville
North Carolina
Digitz — Raleigh
K&S Newsstand — Clemmons
K&S Newsstand — Kernersville
K&S Newsstand — Winston-Salem
News Center — Cary
Parker News Inc. — High Point
Software City — Raleigh
Software City — Charlotte
TDS Computers — Durham
Triad Computers — Winston-Salem
North Dakota
Computer Associates — Fargo
G. F. Computers — Grand Forks
Software House — Fargo
The Computer Store — Minot
Ohio
B5 Software — Columbus
Computer Network — Newark
Computer Site — Cleveland
Earthrise Micro Systems — Columbus
Eddy's Cigar Store — Elyra
Fairborn Home Computer — Dayton
Lakewood International News — Lakewood
Lakes Consumer Electronics — Akron
Leo's Book & Wine Shop — Toledo
Microtech Computer Center Inc. — Toledo
Microware Magic — Fairfield
News Shop — Kent
North Coast Programming — Willoughby
Paper & Ink — Tiffin
Software Centre International — North Olmsted
Software and More — Cincinnati
Software Connection — Mayfield Hts.
Software Plus — Columbus
Somers Photo & Comp. — Cleveland Heights
Specialist Computer System — Toledo
Oklahoma
Little Professor Books — Stillwater
Merit Computers — Oaklahoma City
Micro Link Computer Store — Tulsa
P C Tech — Tulsa
Second Hand Sofware — Oklahoma City
Video-Comp Inc. — Lawton
Oregon
Andromeda — LaGrande
Clackamas Computers — Beaverton
Clackamas Computers — Clackamas
Comm-Shack — Salem
Computron Business Systems — Portland
PC Express — Eugene
Powell's Tech — Portland
Rich Cigar Store — Portland
Sixth & Washington News — Portland
Software Pipline — Beaverton
Tech Books — Portland
Pennsylvania
Alpha Omega EBE — Warren
ATD Software — Sayre
Basic Computer Systems — Erie
Basic Computer Systems — Hermitage
Circle Computer — Ephrata
Computer Solutions — Newtown
Computer Wurx — Stroudsburg
Computers and Games — Reading
Downington Computer Center — Downington
East Coast Software — Hershey
Floppy Disk — Mechanicsburg
Floppy House Software — Palmyra
Gene's Books — King of Prussia
Micro Master Inc. — Monaca
North Central Digital Systems — Danville
Pittsburgh Computer Store — Pittsburgh
Software Hut Inc. — Philadelphia
Tesco Computer Center — Lancaster
Triangle Computers — indiana
Walden Hill Images — Berlin
Rhode Island
Micro Limits — Smithfield
Micro Limits — Greenville
Software Connection — Warwick
South Carolina
Avcom International Inc. — West Columbia
Micro Computer Depot — Sumter
Watson's Jewlery — Chesterfield
Software Haus — Charleston Hts
South Dakota
Computer Business Center — Rapid City
Tennessee
ComputerLab of Memphis — Memphis
Computer Hardware Services — Nashville
Data International — Hickson
The 64 Store — Bartlett
Texas
All Systems Go Software — San Antonio
Amazing Computer Systems — Carrollton
Bookstop — Houston
Bookstop — San Antonio
Byte Shop — Austin
C.E.M. Corporation — Brazoria
Colonel Video — Houston
Computer Age — Houston
Computer Age — Dallas
Computer Magic — Austin
Computer Skills — Eules
FNS Computer Services — Texarkana
Maxwell Book Inc. — Desoto
Metropolitan Computer — Richardson
Micro Search — Houston
Software Library — Witchita Falls
Software Terminal — Fort Worth
The Computer Experience — San Antonio
The Computer Stop! — Amarillo
The Federated Group — Mosquite
Utah
Magazines and Gifts — Salt Lake City
People's Computers — Provo
Vermont
Ray Supply Inc. — South Burlington
Virginia
Family Computer Center — Fairfax
Family Video & Computer Center — Newport News
I/O Computer — Norfolk
Software Central — Norfolk
Software City — Richmond
Tract Software — Prince William
Virginia Micro Systems — Woodbridge
Washington
Amazing Electronics — Spokane
A.P.P.L.E. CO-OP — Renton
Apple Valley Computers — E. Wenatchee
Bellingham Computer Center — Bellingham
Bits, Bytes and Nibbles — Spokane
Bull Dog News — Seattle
Byte World — Kent
Computer Exchange Services — Bellingham
Computer Quest — Edmonds
Computers + Inc. — Tacoma
Family Computers — Everett
Micro Age Computer Store — Bellevue
Nybbles & Bytes — Tacoma
Omni International — Seattle
Port Book and News — Port Angeles
Programs Plus — Seattle
Real Time Systems — Seattle
Take A Byte — Kennewick
Tower Books — Bellevue
Y Not Magazines and Boks — Federal Way
Wisconsin
20th Century Books — Madison
Computer Software Center — Milwaukee
Fox Valley Personal Computer — Oshkosh
Madison Programmer's Guide — Madison
Mayfair Computers — Milwaukee
Mom's Computers — Stevens Point
River City Software — La Crosse
Software City — Racine
TMW Software Inc. — Wausau
Wyoming
Byte Emporium — Laramie
Computer Works — Laramie
Crazy Mountain Electronics — Cooper

Canada
Compute or Play — Alberta
Computerworks — Alberta
Computer Shop — Alberta
Data Pro — Alberta
Computer Shop of Calgary — Alberta
The Software House — Alberta
Hindsin Computing — Alberta
Red Deer Micro Systems — Alberta
Memory Land — Alberta
Compu Care — British Columbia
Computer King — British Columbia
Corel — British Columbia
Nutec — British Columbia
Pacific Business Technologies — British Columbia
Sprta — British Columbia
Strider — British Columbia
Super Software — British Columbia
Canadian Computer Era — Manitoba
Micromart — Manitoba
ComputerWorld — Manitoba
J.W. Allen — Manitoba
Kobetak — Manitoba
Minicomp — Manitoba
Arkon — Ontario
Chasam Computer — Ontario
Computerland — Ontario
Compec — Ontario
Electronics 2001 — Ontario
Hastings Data — Ontario
The Computer & You — Ontario
Thornhill — Ontario
Software Co. — Quebec
Dextronic Solare — Quebec
Mercotec — Quebec
Info Metsier — Quebec
Laval Micro — Quebec
Vision Tronique — Quebec
Distribution Vaipro — Quebec
Distributed in Canada by
Ingram Software Distributors — Concord, Ontario
Compulit Distributors — British Columbia
Phase 4 Dist. Inc. — Calgary, Alberta
Argentina
Logic Computer — Buenos Aires
Australia
Computermart — Leederville,WA
Diskworks — Greenwich, NSW
John Vanhoeven — Cloverdale, WA
Maxwell Office Equipment — Abbotsford, Melburn
Run Soft — Melbourne
France
A.M.I.E. — Paris
Saudia Arabia
Abdulla Fouad & Sons — Daman
Scotland
The Amiga Center of Scotland — Edinborough
Sweden
Nicholas Peerson — Lund
W. Germany
Greenville Trading & Import — Oberursel

Amazing Computing™ is also available in most B. Dalton Booksellers, B. Dalton Software Stores, Crown Books, Software Etc., selected WaldenBooks Stores, and Walden's Software Store locations.

A M A Z I N G M A I L

This edition of the Amazing Computing letters is entirely concerned with "Roomers". We felt the overwhelming response by our readers made this a necessity. The letters below give a truthful proportion to the pro and con reaction of our readership. Along with the overall support for the column, there have been two negative responses (the most eloquent response has been reproduced in this section).

We welcome your comments and suggestions. We hope to find a suitable end to this matter which will maintain the integrity of our "Roomers" column while allowing a level of fair response for the individuals mentioned in each issue.

Since responding to each letter separately would be awkward, we have taken a longer view. A general response and possible solution is the subject of this issue's editorial.

Dear AC:
After following the "ROOMERS" column *debates* since day one, I've decided to put my two cents worth of comments on the table.

I think the column is absolutely wonderful and I salute the BANDITO for his covert investigative skills (any job offers from the CIA or NSA yet ?). Roomers provides exciting, insightful, speculative, controversial reporting and gives your magazine a unique look amongst the predominantly boring crop of computer tabloids. It also reinforces the reader's image of AC as being **FOR** the reader and not being intimidated or corrupted by the developers in the AMIGA market. I've had quite a few enjoyable laughs reading ROOMERS rebuttals from overly sensitive developers crying "foul", making excuses, and threatening legal action (oooooo- aren't you scared?) unless retractions are printed and apologies issued.

From the request you published in the last issue (October) for reader comments on the ROOMERS column's "controversy", I sensed that maybe you were thinking of buckling in to the commercial pressure and terminating the column. Please assure me and the rest of the readership that you will not ! I won't threaten you with anything foolish like subscription cancellation (because the rest of the magazine is very good also), but I will tell you that if you do shut off the BANDITO, my respect for AC will drop by several notches.

Sincerely,
Anthony DaSilva Jr.
Clay, N.Y.

Dear AC:
Re the Bandito. My personal observation is that, however elaborately rationalized, the intent of many manufacturers is to deceive. We subscribers and consumers need your help and that of the Bandito. When a mistake is made, a simple retraction and correction should suffice. I urge you to continue *Roomers*.

If anyone is looking for a good Bridge program, I recommend *Grand Slam Bridge*. It is IBM compatible so a Bridgeboard is necessary. It is available from: Baron Bridge Supplies, 151 Thierman Lane, Louisville, KY 40207, 1-800-626-1598.

I should like to say a good word for *Word Perfect*. They have the best customer service I have ever encountered for any product of any kind.

Sincerely
Hubert C. Minard

Dear AC:
As a loyal reader and recent subscriber, I would very much like to have my say as to the "Roomers" column. Frankly, I love it. I subscribe mainly because of it, and would be incredibly disappointed if you were to discontinue it. You are one of the very few magazines willing to risk such a column and I hope you will continue it.

The only complaint that I could possibly have with your magazine, is that you have elected to divide your articles across several pages, e.g. please turn to page 26, from page 9.

Sincerely,
Lars Benton
Laguna Hills, CA

Dear AC,
I am writing to you regarding your "Roomers" column. I read a great deal of criticism about this column and the "Bandito". I also noticed that the sour attitudes were from developers that feel that their products have received unfair treatment.

Well, I for one, would like to offer my support for the "Roomers" column. As you have stated over and over again, this particular part of the magazine is offered as ENTERTAINMENT. One of the primary reasons (among others) that I purchase your publication is to read the Bandito's latest gossip. I know that the information provided by him/her is not substantiated and I believe most, if not all, Amiga-Owners/ Amazing-Readers, know this also. Even given that many of the rumors printed are not supported, the fact remains: This is for fun.

Please do not let a few sour-apples spoil the fun for the rest of us. Continue the Bandito's column without any restrictions or modifications. Don't give into any developer's pressure if you can help it.

Thanks for the fantastic magazine!

Hank Macklen
Bedford, Mass.

Dear AC,
True story: Today I went to the local bookstore to check for new Amiga mags and was glad to see the new Amazing Computing on the shelf. The cover looked kind of familiar though I thought that maybe I had seen this issue before, so I turned to the Roomers column by the Bandito. I knew that I could tell if this was an old issue because I would recognize the Bandito's column. The first thing I always turn to is Roomers by the Bandito. I actually get excited about getting the new issue of AC because I eagerly look forward to the Roomers column. I noticed that Roomers has grown to 3 pages and I think that's great.

Why I think Roomers is great:

-It has a lively, candid writing style with a great sense of humor.
"TENpoint0.....The Bandito figures they

EXPLORE THE NEW WORLD

Artificial Intelligence
Expert Systems

You've heard the words. You probably know a bit about what they mean, too. You might associate them with white coats, Russian accents and shiny silver robots. What you might not know is that you can explore AI on your Amiga.

MAGELLANTM by **Emerald Intelligence** offers performance and interface features UNAVAILABLE on the IBM-PC*, PS/2* or Apple Macintosh*. It was designed, developed and tested on the AMIGA for the AMIGA.

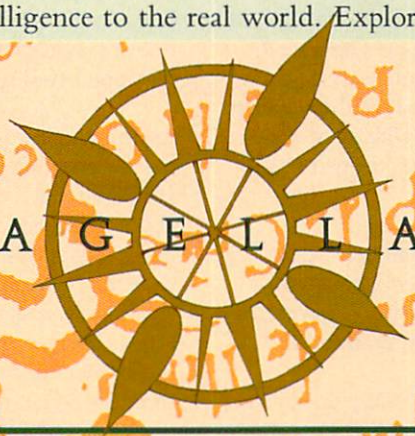
MAGELLAN turns *your* AMIGA into a powerful knowledge engineering workstation. It simplifies some of the complexities of artificial intelligence by using the power of the AMIGA. It is advanced enough to be simple.

The AMIGA's unique capabilities to support inexpensive desktop video, desktop publishing, audio/voice synthesis, extended memory, networking and high-resolution graphics make it ideal for advanced artificial intelligence development. Advanced to the state of being useful, not a graduate-school curiosity. Advanced enough to interface with the real world, and get to work. Advanced enough to be useful.

Use **MAGELLAN** to capture the expertise of your best salesperson, to help junior salespeople "learn the ropes". Build a system to diagnose sick AMIGA's and prescribe repair procedures. Weigh purchasing options in your job or at home. Evaluate performance of the hockey team. Drive animations with rules of behavior of the "actors" or objects. Create original computer art and music. Add real artificial intelligence to games to take them to the edge.

Best of all, use **MAGELLAN** to write programs. No BASIC, no "C", no compiler, no linking, no waiting. No kidding. With programs written with rules, changes can be made interactively — like BASIC — but much, much more efficiently. People don't think in flow charts, or "C" language structures or little curly brackets or parentheses. People think in terms of "IF — THEN". As a child your mother would say "If you take that cookie, then I'll . . ." Rules like this are how people work, how people decide. This is how computers will be programmed by our children.

Welcome artificial intelligence to the real world. Explore **MAGELLAN.**



List Price
$195

Emerald
Intelligence

334 South State Street, Ann Arbor, Michigan 48104. (313) 663-8757

*IBM-PC, PS/2 are registered trademarks of International Business Machines, Inc. *Apple Macintosh is a registered trademark of Apple Computer Corp. *AMIGA is a registered trademark of Commodore Business Machines, Ltd.

went through that many versions of a real name, then gave up."

-It gives us insight into the computer biz. "It also seems as though Aegis's attempt to enter the Mac market has collapsed: Vultures are circling, and buyout rumors are hot and heavy."

-And of course , new product info. "The Bandito's informers in Sacremento report that DeluxeVideo II should arrive early next year. It will finally support all the different graphics modes, and substantially improve the power and flexibility of presentation."

Roomers is chock-full of these delightful morsels of information.

I understand the problem that companies have when they are the object of these rumors. I suggest that these companies give the public credit for understanding the difference between rumors and 'facts'. I also enjoy hearing their responses when they feel that they have been wronged by the Bandito. Any insight into the Amiga developers' world is interesting. If these companies are not satisfied with the space for their replies in Amazing Mail then you should point out the service you do for them hyping their products in Hot On The Shelves.

I hope that you do not put any restraints on the Bandito because of their pressures. That would be like putting David Letterman on in the morning, like making a new formula for Coke, like cancelling Star Trek. (You know many of the loyal ST fans still don't watch NBC.)

Don't be rash- listen to the complaints and give them their space but remember- WE LOVE THE BANDITO!!! I hope that you get tons of mail in support of him (or her). I would like to have the time to write to each of the whiners and scold them for their lack of understanding. Roomers is Roomers and we the readers understand that (even without a disclaimer). I mean geez, really! Cut the Bandito some slack!

I believe in putting my money where my mouth is so I've enclosed a check for a subscription. I am offering this support so that you will support the Bandito.

Very truly yours,
Eric Donaldson

Dear AC:
I have enjoyed your magazine and congratulate you on a job well done. I have recently purchased a subscription but am dismayed by the letters to the Editor in the first issue of my subscription. I am speaking of the letters concerning the column entitled "Roomers".

One of the main reasons I chose to read and to subscribe to your magazine was the presence of the "Roomers" column. There are numerous magazines that purport to examine the Amiga but they all seem to spend their time hyping the various companies that would give me the pros and the cons of programs and hardware and report these things honestly.

While I will readily state that your article has as occasional pro-industry slant, I have come to trust your reports and reviews more than other magazines. The "Roomers" column often tells the possible bad side of the industry and tells which products are over-hyped. It doesn't matter whether these rumors are true, the important thing is that you had the courage to print them (and the wisdom to make certain everyone knows they are just rumors).

I can see why the rumors upset the executives from the computer industry but I believe they are over-reacting. I will buy a product because it appears to be what I want. I could not care less what the management of that company does or does not do in its packaging and development of products. The only effect the "Roomers" column may have on my buying habits is to plant the name of a company or product in my mind and cause me to give that company or product some added consideration (usually favorable) when it is time to buy. To paraphrase the old saying, it doesn't matter what they say as long as they spell your name right.

As an attorney, I am only too aware of the weight rumors should be given. I feel confident that, thanks to your disclaimer at the beginning of each article, your other readers also read these rumors as entertainment and perhaps find them to show your magazine's willingness to admit that the computer industry is still run by human beings. We all do strange or unintelligent things and I am glad the computer industry is the same.

In short, please continue to run the "Roomers" column and congratulate the Bandito on giving your magazine a high grade in truthfulness by freely finding rumors.

With kind regards, I am

Very truly yours,
W.T. Geddings, Jr.
Manning, South Carolina

Dear AC:
In response to your call to keep involved and write comments to your publication, I have to cast my vote for the Bandito, and the "Roomers" column.

I have talked my sister into buying an Amiga 500, but showed her the "Roomers" column wherein a predicted new Apple IIGS plus was/is going to seriously threaten the future of the Amiga. I also told her that the information in the column was of variable significance and accuracy. The charm of this regular feature is well written speculation, which is periodically proved out. The strength of this "unconfirmed and printed for entertainment value only" material I feel lies in its potential for keeping everybody in the Amiga community on their toes (read honest). Keep the Bandito.

As for the rest of the magazine, your scope is impressive. Keep up the good work.

Sincerely,
Ward C. Martin
San Diego, CA

Dear AC,
I have been a subscriber to AC almost since the beginning and have enjoyed the magazine very much. I am writing in regard to the "Roomers" column debate.

I, personally, read "Roomers" for "kicks"—waiting to see which rumors will later be proven true or false. I understand—before reading a word of it—that none of the statements have been confirmed and that much of the column simply reflects the Bandito's own opinions.

In its present format, it is imperative that the AC disclaimer preface the column, so that readers (particularly first-time readers) will not misinterpret the information as fact-based reportage.

However, there will always be readers who consider ANYTHING that has reached print to be FACT, even with the presence of such a disclaimer.

For that reason, it may be wise to eliminate any statements that could potentially damage the reputation of an individual or company until such statements can be proven true. After all, the Amiga community should not set out to discourage developers and related companies from producing new Amiga software/products by hurting them before they have had a chance to demonstrate their intentions.

By the same token, a publication in your position has (or should have) the reponsibility to inform its readers of any product that is inferior, over-priced, or hyped-up and should also notify readers of companies who have mistreated or deceived the consumer in any way.

That is where AC must cross the fine line between loyalty to its readers versus loyalty to its advertisers. Understanding fully that you cannot publish a quality magazine without extensive advertising revenue, crossing this line certainly puts you in an unfortunate position. However, I have always felt that a magazine's first responsibility is to its readers. After all, without readers, you don't need advertisers anyway, right?

You mentioned that in trying to confirm rumors before printing them, you often receive no response or , worse, a denial "followed by an explanation which restructures the original information to be favorable to the company."

In my experience, refusing to deny or at least acknowledge a rumor is as good as saying it is true. As far as a company restructuring the information, that, of course, would be unacceptable to your readers.

Perhaps you could borrow a solution from another publication (non-competing) who found themselves in a similar situation. Instead of risking offense to their advertisers by printing an unconfirmed rumor, they'd print both sides of the story. First, they'd list the rumor and its source (not an actual name but something like "a leading developer said"). Next, the person/company-in-question would be contacted and given a chance to make a statement. Then, both the rumor and the company's statement (or indication of their refusal to make a statement) were printed in the same issue. This gave the reader a chance to judge for himself who was telling the truth. It's not hard to read between the lines and discover a "restructuring of the facts".

I'd hate to see "Roomers" cut altogether because I find it amusing. However, if it becomes more trouble than it is worth, by all means get rid of it....and replace it with a column that reports problem software/products/companies that readers should stay clear of!

Irene Kobelski
Colchester, CT.

Dear AC:
Let me say that number 1, the BANDITO writes one of my favorite columns!! I know enough NOT to take everything as

RETRACT THE DANDURAND STATEMENT!

...or I'll sue.

By Lewis Tilley

Dear Amazing Computing,

I demand a retraction! This may develop into a case of libel! Your magazine printed an insulting article by one of the Durand-Durand boys entitled "The Kideo Tapes" (September issue focusing on "Teaching with the Amiga). This letter is to inform you that I, Lew Tilley, am NOT an elf!... especially I am not " a jolly, old elf" as stated by someone calling himself Dandurand.

You should have known that anyone capable of making a pun like Kideo when referring to videos made by children would be capable of any number of other insulting crimes of desecration.

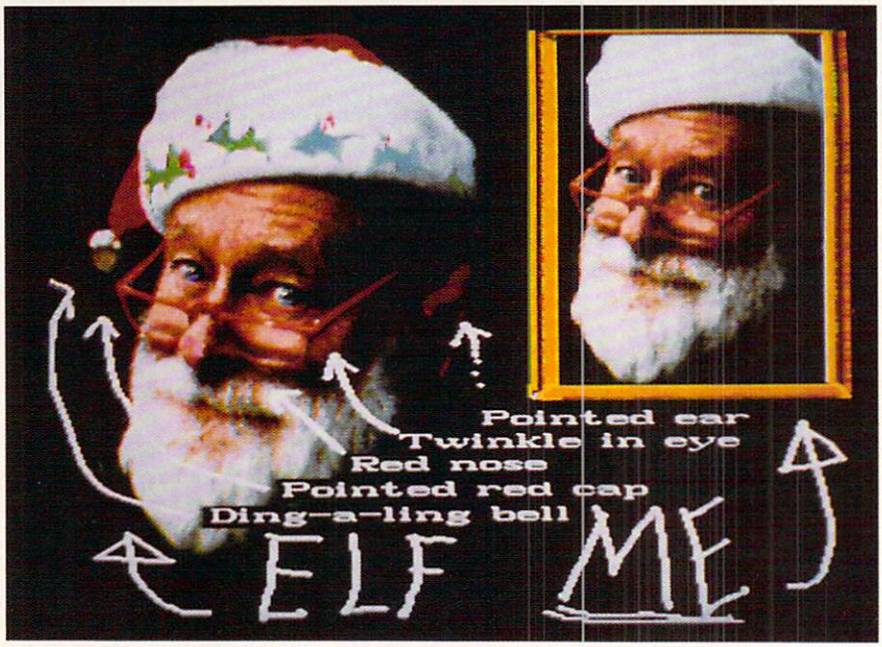
My international reputation as a serious writer, artist and lover has been smeared with a brush and a scurrilous name which may never be removed. In your position as the editor of the "Original AMIGA Monthly Resource", surely you must be aware that anyone who writes regularly for such distinguished magazines as YOUR AMIGA of London, England and a leading Swedish computer weekly; a writer who even now is developing an article on Video for the German Editor, Ulrich Brieden of Markt & Technik Amiga, must maintain an image of impartial hostility toward all software reviewed and discussed.

How can I now maintain this attitude so ably pioneered by men like Ben Dunnington of INFO when I have been labeled "jolly?"

The use of the term "illustrator" may be justified since I was, many years ago, the winner of a Limited Editions Book Club award for my illustrations to the old Thomas Wolfe's "Look Homeward Angel"; and I have to my credit many sets of illustrations for various magazines, school texts and television and film productions. It is the "retired" before the illustrator to which I object.

This Christmas, JUMPDISK is to present Charles Dickens' "A Christmas Carol" illustrated by Lewis Tilley in a special disk edition. Now, is that a "retired illustrator?

Humph! Why can't these young whippersnappers who make the kind of foreign rock music the Durand boys make stick with things they know about? Messing around with loud music the way that they do, maybe they just don't hear too well. "Emeritus" doesn't mean "retired". I am, I'll admit, an "emeritus professor" from the University of Southern Colorado in Pueblo, Colorado. Emeritus really means in its Latin sense "without merit", that is to say, "without reward...or, not getting paid anymore." That, alas, is true, but I do get a free office AND secretarial assistance (when available) to continue all the more pleasurable activities of a full professor of fine art, except teaching classes, from this exciting new technical university in Southern Colorado.



Mr. Tilley clearly illustrates the obvious error made to his character

"Old elf" - that really did it! Have you ever seen a young elf?... of course not. It is this final insult which makes me seek redress. To people who deal every day with eight-year-olds, anyone over the age of thirty must seem "old". If this Durand man, who can't even spell his own name correctly, continues to slander me in such a fashion, I shall be compelled to enter direct competition with him in the production of Kiddyos which feature elves who never retire, are always young, and are never, never jolly.

Signed,
Yr. humble, obedient, elf...er, servant,
Lewis Tilley

Photographs by Laura Audrey

100% "gospel" and to take the column as what it is meant to be - entertainment. Most of your readers also have the good sense to realize that most of the information is based on rumors and speculations, but if you read Bandito's back columns it is very apparent he has been "right on" a very large percentage of the time. I hope the recent pressure does not succeed in eliminating the "ROOMERS" column. His column, like your magazine, provides a ' breath of fresh air ' to AMIGA owners who quickly tire at the lack of skepticism at some other publications. We need honesty and objectivity not another "house organ". I have never felt "THE BANDITO" was being malicious in any of his columns and feel that some of the complaints against the column have basically been "MUCH ADO ABOUT NOTHING!" In summary, let me say- "LONG LIVE THE BANDITO"!

Earl Davis
OHIO

Dear AC:
I feel the "Roomers" column should be left just the way it is. I personally love The Bandito's writing style, and I find the column a substitute for all the chatter that goes on at the trade shows which I cannot attend.

The magazines responsibilities in this area are : 1) to state explicitly that the column is for unsubstantiated rumors; and 2) to only publish material after attempts at clarification and /or substantiation have failed. According to your comments at the end of *Amazing Mail V3.10*, you DO perform obligations.

The developers and publishers are as much to blame for the proliferation of false rumors as is the Bandito! If they don't want products announced, or want to keep details of their product secret, then it is THEIR responsibility to keep their people quiet until the proper time. The Bandito hears all this stuff from somewhere!

Also, they should look to this column for information on how the Amiga community may have an inaccurate perception of their product. When they see a rumor listed in the column, instead of lamb-basting the Bandito, they should be glad someone "caught" the rumor, and they can then put it to rest.

Let's keep the Bandito, and remember the first amendment—as long as it's handled responsibly.

Derek Buckley
Spokane, WA

Dear AC,
I received your October issue in the mail today (thank you for putting the month on them). I wonder if it is coincidence that the issue with three letters of complaint against the Roomers column is also one of the funniest Roomers columns ever. For instance, a certain software package "had more bugs than the American embassy in Moscow, was as slow as Jack Tramiel reaching to pick up the check for lunch...".

By now you may have guessed that this is a letter in favor of Roomers! Roomers, however, is my favorite column. I read it first, then the rest of the magazine. I can understand that no one wants to see their name or product associated with bad publicity-and rightly so. But that is no reason to write threatening letters along the "my attorney will be contacting you" vein.

Your disclaimer states that it is "printed for entertainment only" and it certainly entertains me. Keep it up.

Sincerely,
Michael Carpenter
San Jose, California

Dear AC,
A "Tempest in a Teapot" has struck Amazing Computing magazine in the Amazing Mail section.

It is indeed "Amazing" and amusing to find that certain people feel libeled by rumors in a magazine such as Amazing Computing. In my opinion the "rumors" column is the "National Inquirer" section of the magazine. It is full of half truths and wishful thinking and is there for the entertainment value. I enjoy the column and it is the article I read before all others. But if I based my purchase decisions, whether to purchase now or wait for the rumored product, on information in the column I'd be a fool waiting forever for something that may never see the light of day. If I'm interested in a product I always look for legitimate reviews of the product to base my purchase decision on. I never base a purchase on something written in an amusement column. If a rumor says that an upgraded product will be available "soon", I still purchase the initial issue of the product knowing that reputable companies always have an upgrade policy which I may avail myself of later. In the meantime I've used the basic product and will be ready for the upgrade features when and if they arrive on the market.

I hope more peole learn to dispense with the "tempest" and enjoy the "tea". By the way, when did you say that the Amiga 75,000 was due for release?

Bill Braun
Vallejo, CA

Dear Amazing! folk-
Keep the disclaimer.
Keep the Bandito.
Keep the Roomers department. Please.

I do not read your fine magazine to see what someone in a lawyer suit thinks is ok for you to print. Nor do I read it to have established manufacturers/ developers tell me to postpone my purchases for just a little while because their soon-to-be-released vaporware is truly (honest! trust me!) what I've been waiting for all my life.

I read Amazing! for news, for insight, for opinion, for all your contributions to the Amiga community. And I read it for entertainment, gossip and rumors: The 'flavor' of our industry/community.

Amazing! is unique. Please keep it that way.

Yours Truly,
CapE.B. Schwartz
San Francisco, CA

Dear AC,
One of the most appealing points your magazine has going is the lack of self-proclaimed importance (some blue-suit companies call this "polish" or "commercialism"). Please excuse those in blue suits whose scathing letters may have left burns, they come from a crowd that thinks "tongue-in-cheek" means an announcer stumbled while reading from a prompter.

Please let those who never loosen their ties know that "Roomers", to the layman, is synonomous with "Rumors". The Shorter Oxford defines rumor as " General talk, report, or hearsay, not

based upon definite knowledge." It also lists "Talk or report of a person or thing as some way noted or distinguished", but that definition is listed as archaic. Maybe those who take their products and the stories about same seriously , should try getting in touch with "their public". I asked 22 people who read AC what they thought of the Bandito & if they took "Roomers" seriously. With no exception, the Bandito is an ok kind-of guy, and anyone who gives credence to a column titled Roomers deserves whatever they get. One said he gets as much entertainment from the Bandito's column as from software houses' product announcements, and finds one just as credible as the other. (For those who aren't sure about that, it means a person can't put any stock in either.)

In an industry taken as seriously as computers, it's nice to have some humor now & again. Leave "Roomers" alone.

Sincerely,
B. Gray
Sun Valley, CA

Dear AC,
I'd like to add my input about the "Roomer" column in your magazine. As long as the roomer starts outside of the column, it should be fair game. I take what I read there with a grain of salt. After a user group meeting we find ourselves talking about some rumor that was in the last AC, and asking "Do you think that it might be true?" It is a fun column , as long as it is understood that these bits are indeed rumors.

Keep up the good work!

Scott W. Smith
Sherman Oaks, CA

Dear AC,
I recently heard that your magazine was being forced to drop the roomers column, due to legal matters. Having just purchased the 3/10 issue of your magazine I find info in "Amazing Mail" that appears to support the possibility of such. Being aware of lag time in magazine production and the fact that there was no label of "rumor" or even a disclaimer given by the individual who told me such, I feel there is enough to enquire about.

Being a bit more than a reader of AC (I have a copy of all the issues) even the first issue had a "Roomers" column (without a disclaimer and mentioned Commodore to soon be releasing "Amiga LIVE!" to be available before the end of February->87?). Now it seems to me that rumors get started by even the best of us and the thing to do is check the facts (even the world was flat until the facts were checked out). AC has been doing a great job of giving users a quality magazine and to lose the "Roomers" column would not only be an injustice to AC but to the readers as well.

Rumors are a fact of life and AC does justice by putting them under the heading "Roomers" which is more than most do who create and pass such non-verified information. I hope the "Roomers" column stays, besides rumors are a part of the evolution of the Amiga and a bit of reality would be missing without them. What better place to mention such things as VaporWare while leaving hope for such and perhaps measure reader interest?- enquiring minds want to know!-

(name withheld)
Tucker, GA

Dear AC:
The Metropolitan Symphony Orchestra of Boston acquired an Amiga 2000 this past year. I was responsible for the choice; and while I would like to tell you that I picked the Amiga because I made a reasoned and informed selection based on our needs, the truth is that I was dazzled by the Hi Tech demos. The real reason for my choice was a lust for all the fantastic graph and audio capabilities.

It has turned out that the very qualities which led me to the Amiga, are the features which have proved to be the most important. The world of the Cultural Non-Profit, like that of the software houses, is a Darwinian Jungle where you either outshine, outperform, or otherwise distinguish yourself from the competition, or you go out of business.

Our first software came from Aegis. *Sonix, VideoTilter, Audio Master,* and *Draw Plus,* were the programs on which we learned to operate the Amiga. Since then, we have added nearly twenty additional programs from other companies.

I like the products from Aegis. I found them to be uncomplicated, well documented, and easy to use. They do useful work, and they do it well. Once you have learned an Aegis program, mouse use becomes a joy. The conventions and pull down menus used by Aegis are standard. You know where to look for "print", "save", "open", "new", etc. from program to program; and once learned, it applies to other company's software as well.

We have used *Draw Plus* to design a potential concert hall and to visualize stage settings. *Sonix* allowed our conductor to hear an unknown score before committing himself to a performance; and *VideoTitler* gets lots of use on our sample video cassettes. *Audio Master* opens horizons to us as musicians and acousticians. In short, Aegis has quality products at sensible prices, and in my opinion, Aegis software has helped to put the Amiga in the race as a serious machine.

I am appalled at the cavalier statements made in your "Roomers" column in your October issue. "Vultures are circling" and the suggestion that they are in a state of collapse is irresponsible and destructive journalism. If I were a first time buyer, I would avoid a company described in the language used by your magazine. I want your readers to know that if my experience is any guide, Aegis products are a solid value and deserve to be bought and used.

I do not believe that your wishy-washy response to Aegis in the November issue is sufficient. In the years in which I published as a music critic, no editor in his right mind would have printed anything like the scurrilous language in your "roomers" column had I said such libelous things about a performance group.

As long as there are people like me who continue to buy, use, and recommend Aegis products, their company is secure.

Yours truly,
Arthur Frank Benoit,
Executive Director
Metropolitan Symphony Orchestra
Boston Light Opera Company

Do you have a concern? Don't just sit behind your keyboard. WRITE!!!

From The Managing Editor: Putting "ROOMERS" to rest!

It started innocently enough. I received a telephone call from a member of the Amiga community who wanted to write a monthly column for Amazing Computing. They argued there was a need for an avenue of "leaks" to the Amiga community from Amiga developers. Why not a column dedicated to viewing the inner workings of the Amiga market. In order to get the best information from sources who would not be judged by their association with a known columnist, a pseudonym was invented. "Roomers" was born.

"Roomers" has become an institution (cursed or otherwise) in the Amiga community. The column has seen several authors (each leaving when they could no longer maintain the pace of a monthly column), yet each individual has carried a commitment to providing the Amiga community a penetrating look into the movement of this growing market. In my opinion, each individual worked hard to provide insight and an honest reflection on the expansion of our favorite computer and the individuals involved.

The column has always been well received by our readership. Readers want a hint of the new products or improvements on the horizon. From the letters we receive, our readers maintain a distinction from the factual reporting in Amazing Computing and the rumors presented by The Bandito.

However, the column has remained a bane to developers and advertisers. Through phone calls and individuals cornering me at Amiga conventions, some Amiga developers have let me know they are not amused. However, few have pointed to factual inaccuracies in the articles. Even the most negative responses have been followed by an explanation of why The Bandito may have misunderstood their meaning and efforts. Almost all have neglected to place their dissension in the form of a letter to be presented in this forum.

One individual challenged me in public by stating his company would not advertise in Amazing Computing while we carried "ROOMERS". I mentioned the rather obvious fact that "ROOMERS" was only one to three pages of a magazine which carried between 112 to 128 pages each issue. Did he feel he was applying a little too much weight to this rather small percentage of the magazine?

No, there was a principle involved. He sincerely felt we were committing a disservice to the entire Amiga community by continuing "ROOMERS". It did not matter how much we worked in every other effort of our publication, he was insistent that we be judged by "ROOMERS" alone. He remarked that our

"Is that all?"

"That's all I can see." I replied.

"But that's not bad, in fact its complimentary. Why would someone tell me that "ROOMERS" said we were closing."

"ROOMERS" column made us the "National Enquirer" of Amiga journalism. To this day we have not received his promised letter to print in Amazing Computing.

Even The Bandito receives an occasional "Bad Rap"

Recently, I received a call from an anxious developer who had been told The Bandito had written some bad things about his company in an issue of AC. The individual openly admitted he had not yet seen the article and wanted to know what was said.

I read the portion of "ROOMERS" which mentioned his organization to him over the telephone. When I finished, he said,

"Is that all?"

"That's all I can see." I replied.

"But that's not bad, in fact its complimentary. Why would someone tell me that "ROOMERS" said we were closing."

Why indeed. Is it possible The Bandito's column is now seen by some readers as a constant negative, disdaining voice? Are the insights in "ROOMERS" being read by individuals who have already made up their minds as to what the column says, and not taking time to read the full text?

Let's be honest, The Bandito rarely pulls a punch, but I have seen many positive statements and compliments written in "ROOMERS". Yet, if a good many people are allowing bias to rule their interpretation, we will quickly perceive everything in "ROOMERS" as negative.

Asking The Right Questions

There are no easy answers. However, the best way to discover the right answer to a problem is to ask the right questions.

To The Readers:

Do the readers of Amazing Computing view The Bandito's comments as the statements of a single individual? Are our readers aware that this individual speaks on the topics and concerns which affects us all from a single lone viewpoint? Do our readers balance their intake of rumors with a good dose of common sense?

To The Developers:

Does the Amiga developer community understand the importance of an alternative viewpoint? Do they see the need to counter the overpowering crescendo of superlatives from individuals whose occupation is to demonstrate the better points of their product, while down playing the negatives? Do they understand the sense of balance that "ROOMERS" gives AC (out of 112 to 128 pages a month only two or three are used by "ROOMERS")?

To The Bandito:
Is our avenging force consistently balanced in the reporting being conducted? Is this person taking the wide view on every issue? Is he maintaining a level of fairness? Is he saying things through the guise of The Bandito, which he would openly say to the individuals he is reporting about?

To Amazing Computing:
Are we doing all we can to maintain an even course by allowing the Bandito complete free reign? Is the fact that these are rumors enough of an explanation to the general public for their existence? How extensively should we monitor this column? When should we contact individuals for their side of an issue? How much weight should we give a negative response from an angered developer?

Getting The Right Answers
While there are no complete answers, as a publication dedicated to serving the needs of the entire Amiga community, we must find a means to keep the integrity of the "ROOMERS" column and still be as fair as possible to all involved. This must be done in such a way as to allow The Bandito the freedom to report, and individuals the right to be free from any unjust persecution.

For this reason, in the future "ROOMERS" will maintain a disclaimer, but we at AC will reserve the right to periodically contact organizations and individuals for responses. While leaving the original column, these responses will appear within the "ROOMERS" column as asides from the editor following their related sections.

In this way, the "ROOMERS" column remains an open forum. Yet, we will be allowed to give a second viewpoint to some of the more controversial statements made within the column.

The Developer's Responsibility
This procedure will only work if the developers concerned return our calls and respond to our questions. We know that they will not readily admit to producing a top secret product, or an inner war between members of their firm, but this will give individuals an opportunity to respond in the same issue and in the same forum as the statements to which they object.

If these individuals have a problem with the written response or feel they were not allowed enough space to respond properly, we will maintain our open suggestion to write the magazine. If they have an alternative view, we want to hear it and print it. Our purpose is to cover the entire market, not just that with which we are most comfortable.

I want to take this opportunity to say a special thank you to all who have responded on either side of this issue. It is through communication we are able to reach deeper within ourselves and farther towards each other. The sense of Amiga community is a major focus of our efforts at AC and this response has shown how involved our fellow Amiga users can become. Thank you.

Sincerely,

Don Hicks
Managing Editor

Hot on the Shelves

by Michael T. Cabral

Slurping Clones

Five life force-hogging alien clones are determined to turn you into a computer. (God forbid!) Helplessly, you watch as your limbs and organs turn into tiny electronic components. Your survival depends solely on your ability to find your twins, disintegrate them, and gain the vital life fluids you desperately need. So the story goes in **Captain Blood**, the latest space-age battle from Mindscape, Inc.

Of course, tracking down your despicable partners won't be easy. They are scattered all over an unfriendly galaxy that could take light years to cover. Perilous flights put your piloting skills to the test, and the terrrains are ragged enough to shred your ship into cole slaw. When you land, things get uglier. The lands you comb are infested with unspeakable aliens. Tangling with Migrax, Croolis-Ulv and the sinister seductress, Torka, makes your dwindling life that much more difficult.

Your final enemy is time. An on-board clock ticks down the frantic two-and-a-half real-time hours you have to find some of that crucial life fluid. Snag one of the clones and you are awarded another two-and-a-half hours of desperation and sweat. You also have a control panel and a mechanical arm that, sadly enough, is your own. With this arm, you make your desperation moves. And as time slips away, the arm begins to shake, a victim of your degeneration.

Captain Blood also adds some fresh twists to the graphic adventure genre with a unique icon transmitter language and a whopping 32,768 planets to explore. Throw in advanced 3D flight simulation, dazzling graphics, and some racy humor and you've got a space adventure that tests every aspect of even the sharpest space travellers.

Captain Blood $49.95
Mindscape, Inc.
3444 Dundee Rd.
Northbrook, IL 60062
(800) 221-9884

Preferred Preferences

Slavery to Preferences is a liability all Amiga users face. Want to print a file or picture? We all know the routine. Trudge into Preferences, make all necessary, tedious changes, then you can finally print. Maybe. If you haven't made some minute error that forces you back to Preferences for another run around the circle, you are alright. Otherwise, get ready for the runaround.

Soft-Link Inc. is out to end Preferences frustration with **Multi-Prefs**, a utility that allows you to preset multiple Preference settings. You can create settings for your specific situations, name them, and store them for later use. Whenever you need those settings that would have been such a bother to reset, just call up your convenient Multi-Prefs premade settings.

Multi-Prefs also lets you combine portions of certain settings with parts of other settings. If you need the colors from one setting, the printer information from another, and the mouse pointer from a third, Multi-Prefs is cooperative. For complete control of Preferences, you can also edit, sort, rename, delete, or undelete your settings.

In addition to the Multi-Prefs utility, the Multi-Prefs package includes three other goodies. "MP" is a CLI version of Multi-Prefs, allowing you to restore full or partial Multi-Prefs settings from the CLI. "Up" advances printer paper to the top of the next page with the click of an icon. "Lace" toggles your Workbench screen between interlaced and non-interlaced modes.

Multi-Prefs
Soft-Link, Inc.
P.O. Box 304
Coventry, RI 02816

PostScript Print Perfection

If one good utility deserves another, then New Horizons Software answers the call. **ProScript**, a PostScript print utility, brings typeset quality printing to your Amiga. The program translates ProWrite documents to the

(continued)

Captain Blood

PostScript language for output to a laser printer or other high-quality print device.

For the PostScript uninitiated, ProScript lets you scale fonts, texts, and graphics to any size on your Amiga. Since PostScript fonts are defined by filled lines and curves, the characters escape "jaggies" and the "dotty" look of bitmapped fonts. PostScript is also fast and device independent. Whether you are printing on a Linotronic or a basic dot-matrix printer, your PostScript file works fine.

ProScript enhances the PostScript capability with other useful, related features. For instance, PostScript fonts can be automatically substituted for non-PostScript fonts, or you can download non-PostScript fonts to your printer. In the WYSIWYG tradition, PostScript pages print precisely as they appear on-screen. ProScript also allows you to save the PostScript file to disk, in lieu of printing immediately. And for ease of transition for ProWrite users, ProScript's user interface is similar to the ProWrite interface.

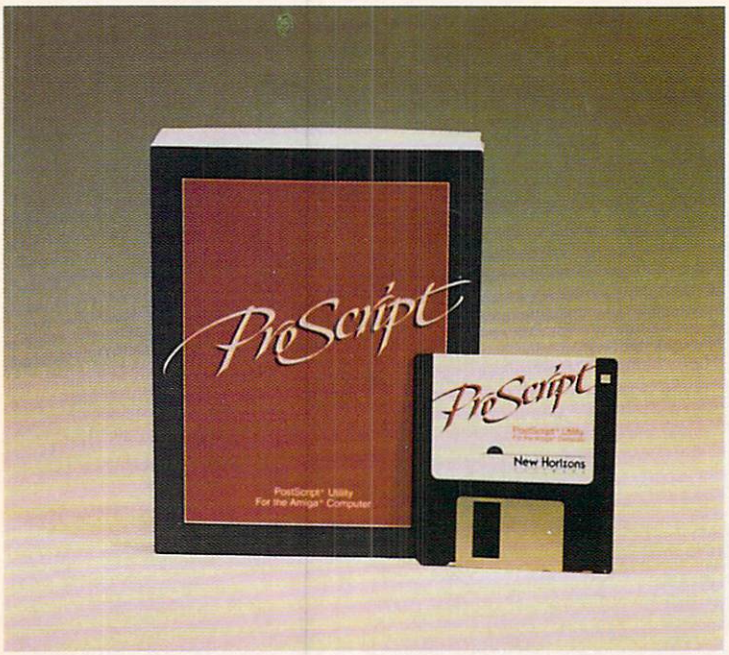
ProScript
New Horizons Software
P.O. Box 43167
Austin, TX 78745
(512) 328-6650

Photon Animation

As Microillusions series of Photon products continues to grow, so do the demands on the creativity of Amiga users. Once upon a time, a stunning still drawing or a revealing framegrabbed image were strokes of creative genius. Now with the **Photon Video: Cel Animator**, you are dared to spring any and all of your Amiga images to life! The latest addition to the Photon family allows you to sequence live action animation frames captured by a frame-grabber, paint or draw a series of frames, or even create stop-motion animation with a video camera and digitizing software.

Cel Animator frees you to mold your animation at all stages. With the program's Pencil Test option, you can preview crystal clear black-and-white video camera scenes and check the motion on your Amiga screen. From there, you can select your playback speed per second and set on-screen delays. Your scenes loop automatically, so you are free to polish your work without the nuisance of constantly rewinding. Timing is also under your control with a simple switch of the display time. The Pose Test option lets you adjust your timing without reshooting. Once you've got the timing roughly where you want it, you can add breakdown drawings, and re-time your delays for pinpoint accuracy.

Cel Animator also throws off the constraints of sequential media. Rather than forcing you to shoot and reshoot film or video scenes in exact desired sequence, the program lets you shoot each image only once. As you record, each frame is tagged with a number, so you can move frames around, change your order completely, and add or delete frames without the tedious work of reshooting. With Cel Animator, useless backtracking and reshooting are not parts of the editing process.



Cel Animator also recognizes that your animation is only half complete without sound. The program lets you digitize and save an audio track with any Amiga sound digitizer and then load it into Cel Animator. You can then step your audio through, frame-by-frame in real-time with no distortion to see and hear precisely how your animation will fly. Deciphering sounds and storing phonemes and sound effects according to frame number allows you to print out an "exposure sheet" and track the bits sound in your animation frame-by-frame.

Cel Animator includes many other features to open up your animation options and make your work easy. The program loads images in all 40 Amiga graphics formats, and opens and saves any IFF files. Your playback choices are highlighted by reverse playback, playback with sound, and slo-mo. A box of drawing tools is stuffed with pen select, line, flood fill, circle, color cycle, and more.

Photon Video: Cel Animator
$149.95
Microillusions
17408 Chatsworth St.
Granada Hills, CA 91344
(800) 522-2041

WordPerfect at a Price

WordPerfect for the Amiga may not be new, but WordPerfect Corporation has worked out a new deal for user groups. Through December 31, 1988, the acclaimed word processor is available to U.S. user groups for $155, a far cry from the $329 retail price. Interested users must show proof of Amiga user group membership, and all orders must be accompanied by a WP Corp. user group purchase agreement (available directly from WP Corp.). Can you think of a better reason to latch onto a user group?

WPCorp Amiga Orders
Special User Group Price, $155
315 N. State St.
Orem, UT 84057
(800) 321-4566

•AC•

AmiEXPO California '88
The Amiga Event

by Stephen Kemp and Stephen Pietrowicz

From the moment the doors opened to the public, AmiEXPO California was a sight-and-sound extravaganza. This was the fourth AmiEXPO to be held and the second for California this year. As with each of its predecessors, this show proved to be larger and even more exciting than the last.

Even before the show opened, you could feel the excitement in the air. We arrived at the Westin Bonaventure hotel, in Los Angeles, about two o'clock in the morning. By the time we finished checking into our room, we had already seen several exhibitors and organizers roaming the lobby and halls, waiting for the show to begin. When AmiEXPO opened at noon on Friday, the eager people waiting in the long lines outside the exhibition hall began to pour in through the doors.

The first stop for most people was the NewTek booth. They were greeted by Laura Longfellow, the "face" that you see in many of their demos, including "Maxine" Headroom. Demonstrations of their main products, Digi-View and DigiPaint, were held in the booth, as well as, the new NewTek Demo Reel II. This "reel" is as impressive (if not more) than their first and included a number of sequences from popular sci-fi shows like "Star Wars", "Star Trek", and "Aliens". NewTek was also discussing their newest product, the Video Toaster, which is a real time, full color digitizing, video effects, genlock system.

Most Impressive!

Perhaps the most impressive thing at the show (which is saying a lot since almost every booth had something impressive) was at the ASDG exhibit. Perry Kivolowitz, president of ASDG Inc., announced that their new color image input system, SpectraScan, would begin shipping by the end of October. This software and hardware package provides impressive, state-of-the-art, high resolution, full color image input and output processing. I know that sounds like a mouthful, but it appeared to be everything that ASDG claims and perhaps more.

Driving a Sharp JX-450 color scanner, SpectraScan is capable of inputting a color image up to 11 X 17 inches in size. The input resolution is variable from 30 to 300 dots per inch and will report 24 bits per pixel of color information. That results in a palette of 16 million colors! One of the most fascinating things about this product seemed to be how easy it is to scan an image. Every operation takes place on the screen, in real-time, on the image that you are scanning. A "trial scan" can be selected to quickly input a black and white image of the original, which then allows you to select a subrange to concentrate on and enlarge. The "fine scan" has all the capabilities that the package allows and will input a full color image of the subject material that surpasses anything currently available in the Amiga market.

Now if you are sitting out there with

AmiEXPO California '88 attendees arrive early

The first stop for most people was the NewTek booth

Perry Kivolowitz, president of ASDG Inc., demonstrated SpectraScan

your calculator, you might have figured out that a full 11 x 17 inch, 300 DPI, 24 bit color image would require a tremendous amount of memory. Don't worry, ASDG has addressed this problem by implementing a demand paged virtual memory environment. With as little as two megabytes of FAST memory and a "large" hard disk, the image can be handled quite easily. They demonstrated that scrolling through the image using SpectraScan was actually faster than some paint programs currently available. And there isn't another one available which can handle a picture near this complexity.

"Okay, it handles color input and manipulation quickly and easily, but how about output?" Is this your next question? Well, SpectraScan can convert its 24 bit per pixel data into a standard Amiga HAM image as well as other standard image formats. In fact, future releases are expected to support conversion to popular Apple and IBM image formats. Several "hard copy" images were on display in the booth and most were printed by the H-P Paint Jet printer. This further demonstrated the power of this package.

ASDG believes SpectraScan may change what the words "desktop publishing" means in the future. SpectraScan lists at $995 for the software and hardware interface and the Sharp JX-450 scanner lists for $6995. Although it is not inexpensive, if you are considering entering the professional publishing or merchandizing markets then you better not overlook this offering from ASDG.

Another offering from ASDG (which is included in the SpectraScan package) is the Twin-X general purpose I/O board. This is an Amiga 2000 compatible expansion card which can host two standard IEEE 959 modules or one double- wide IEEE 959 module. A standard IEEE 488 interface module, available from ASDG, can be attached to the Twin-X board which will allow your Amiga to communicate with a number of mechanical devices including, test equipment, plotters, and medical equipment. The Twin-X retails for $329 while the IEEE 488 interface module costs $199.

Moving around the floor, it was obvious that many attendees spent a lot of time drooling over the hardware expansion booths, trying to determine who was offering the most hard drive or memory expansion for the money. Great Valley Products (GVP) was one such exhibit. At GVP's booth everyone had the opportunity to view hard disk drives and controllers available for the Amiga 500 and 2000, as well as, FAST RAM cards. Amiga 500 owners may want to make note that GVP's Impact A500 hard drive system provides an internal connector for installing up to 2 megabytes of additional memory. For the 2000 owners, a "HardCard" may be just what the doctor ordered. GVP offers a hard drive on a card, available in 30MB or 45MB sizes. The advantage of the HardCard is that the expansion bay is left open, making it available for other peripherals.

MovieSetter, by Gold Disk, is billed as the first true WYSIWYG Amiga animation program

Supra Corporation was on hand with a full line of Amiga peripherals including hard drives, memory boards, and 2400 baud modems. The drive systems are full SCSI compatible and available in sizes up to 250MB. Memory boards containing up to 2MB can also be included into the drive systems available for the 500. Supra representatives were demonstrating just how quickly a series of IFF files could be read from the disk, achieving an almost "animation" capability.

Three new boards were shown at Interactive Video Systems' exhibit. They introduced a new multi-function board called the Grand Slam for the Amiga 2000. This board includes a SCSI hard disk controller, parallel port, serial port, and can contain up to 8 megabytes of RAM. What more could you want? Attendees who didn't need all the capabilities of the Grand Slam were shown IV's new hard disk controller, TrumpCard. The TrumpCard is a new entry into the low-cost SCSI disk controller market, retailing for $189. It can support up to 7 drives chained together, and has a socket for the new 1.3 Auto Boot ROM. Interactive Video didn't stop there. Those waiting to "build" their own boards were introduced to Prototyping Tools, a fully functional, auto-configurating, bareboard for the Amiga 2000. With these three offerings Interactive Video can certainly say they have something for just about everyone.

Spirit Technology was on hand to demonstrate a 2 megabyte memory board for the Amiga 500. This is an internal expansion board that fits under the radiation shield in the ventilation air flow. It uses 1 megabit 256k x 4 DRAMs and requires no jumpers and no soldering for installation. Also demonstrated was a ST-506 hard drive adaptor for the A500. This adaptor (also available for the A1000) attaches to the expansion bus and provides a slot that can hold a standard ST-506 hard drive controller (half size card). The ST-506 interface is compatible with a wide range of hard drives found in the IBM and compatibles marketplace.

Memory And Storage Technology (MAST) was present demonstrating their unique hard drive (Tiny Tiger) that interfaces through the parallel port. That's right, the printer port! It incorporates something they call "Transparent Transfer" circuits that allows it to distinguish between disk I/O request and other parallel port functions. MAST also had a few other of their "Matchbox Collection" on hand, which included a dual floppy disk drive set that is about the size of the standard Amiga external drive box.

Want to speed up your A2000? Computer Systems Associates was on hand to demonstrate several products that would do just that. CSA offers accelerator boards for the A500, A1000, and A2000. These boards are designed to replace the 68000 with a more capable 68020 running at 14mhz. This can cut the time required by CPU intensive programs dramatically. If you have an A2000, CSA is offering the DragStrip which can pack up to 16 megabytes of memory and features lightening fast screen updates and hard drive accesses.

For those of us with A500s and A1000s, who want the A2000 capability without giving up our current machines, Comp-U-Save may have just the ticket. The BusExpander, manufactured by Bill's Boards, is a motherboard that can be attached via cables to the A500 or A1000's expansion bus. Once placed inside an IBM AT style case with a power supply, you can then use any of the A2000 or IBM expansion cards that will fit in the A2000. Of course it takes a little work and requires more room than a 2000 would, but it may be a more cost-effective way to upgrade. The BusExpander sells for $495 and an AT case with power supply should cost less than $200.

Naturally, the exhibitors demonstrating genlock capabilities and video titlers drew their share of the crowds. The video interface technology seems to be growing by leaps and bounds due to the Amiga, opening doors to a large number of potential vertical markets.

The people at A-Squared Distributions' booth were hard at work demonstrating their newest hardware for video effects, the LIVE!2000. LIVE!2000 rounds out their line of LIVE! video frame grabbing hardware which has already been available for the A500 and A1000. The

A2000 version has two video input jacks and allows you to fade, wipe, and cut between the two video sources. The list price for the LIVE!2000 board is $450.

Elan Design was on hand demonstrating their software which relies upon the LIVE! hardware. Invision allows the Amiga to become a real-time video effects system. With this product you can change the image color, mix it with an image from your favorite paint program, or stop the action. Anyone thinking about making their own "film", mixing animations with real life, will have to give Elan Design a look.

Several vendors were on hand demonstrating the latest hardware available for broadcast quality genlock. Two such exhibitors were Digital Creations, offering SuperGen, and Magni Systems with their 4000 Series. The products offered by these companies had a number of people mesmerized for long periods of time. Both products look surprisingly similar on the outside, although there were some significant differences. Each product comes with its own "controller" panel which has two slide pots to control the mix of the video signal from an external source and the Amiga. Adjusting one of the slides from one end to the other causes the image on the screen to switch gradually from one source to the other. Magni's controller also offered the ability to mask out one of the colors from the Amiga source allowing the video signal to show through from behind. Perhaps the most significant difference between the products is that SuperGen is compatible with all three Amiga Systems, while Magni Systems' 4000 Series is only available for the 2000 and another computer called the IBM.

Most everyone in the Amiga community knows what a Boing Ball is, but how about the Boing Mouse? What's a Boing Mouse? It's an optical, three button mouse for the Amiga that was demonstrated at AmiEXPO! Instead of a roller in the bottom of the mouse, it has a LED and comes with a mirror pad. The mirror pad has a grid on it that tells the mouse where it is moving. No more "click-click-click" when you roll across the table! The Boing Mouse is compatible with the existing Amiga mouse port and comes with a 4 foot cable. Boing Mouse is slated for release on December 7, and will be available from: Boing, 1881 Ellwell Dr., Milpitas, CA.

If their interest in Amiga Hardware waned, attendees of AmiEXPO had a variety of software exhibits to visit. Of course some of the most popular booths were the booths demonstrating graphics and animation software. A number of new titles were demonstrated at the show or announced for "eminent" release.

Antic Publishing, a familiar name in the Atari arena, is making a big splash in the Amiga marketplace by introducing two new software packages and a disk collection that users of Sculpt 3D and Videoscape 3D might find interesting. Zoetrope, authored by Jim Kent who brought the Aegis Animator to the Amiga, is Antic's new animation system. Zoetrope gives the user the ability to create cell animations over any range of frames. Jim didn't forget compatibility either — Zoetrope accepts the file formats of IFF, VideoScape 3D, Aegis Titler, Aegis Animator, and LIVE!. Antic has also entered the games market by introducing the first HAM mode graphics game, Pioneer Plague! Pioneer is a multi-level, arcade style, strategic action game that uses the HAM graphics mode and stereo sound. This one may be the first of a new generation of video games.

Not to be left out, Byte by Byte was on hand fielding questions and demonstrating Sculpt 3D, Animate 3D and their latest offering, Sculpt Animate 4D. Needless to say, there was always a crowd at their booth watching with fascination.

MovieSetter is another interesting animation product announced by Gold Disk. MovieSetter is billed as "the first true what you see is what you get animation program for the Amiga". Gold Disk says that this product includes animation techniques that were pioneered by Walt Disney studios. It will allow you to create several minutes of "movies" without tremendous memory requirement (1 meg is recommended). MovieSetter also handles animation speeds up to 60 frames per second, and supports full stereo sound and panning. For those of us who aren't artists, a set of

"characters" is included with the package.

By the way, if you have produced an impressive graphics and/or animations demo for the Amiga, then you might be interested in knowing that AmiEXPO has announced a graphic and art contest! Categories and prizes for the contest will be:

Two Dimensional Image - An A-Pro Draw Package from R & DL Productions, Three Dimensional Image - A "Piggyback" Accelerator from CSA Digitized Image - A "Perfect Vision" Digitizer from Sunrize Industries Animation - A 20meg Hard Drive from Supra Corporation Mixed Media Video - "Live" from A-Squared and "Invision" from Elan Design

The contest will be judged during AmiEXPO in New York, March 3-5, 1989. Official rules and application forms for the contest can be obtained by writing to:
AmiEXPO: Art/Video Competition
Attn: Stephen Jacobs
211 E. 43rd St Suite 301
New York, NY 10017

A sight and sound extravaganza

As we stated at the beginning of this column, AmiEXPO California '88 was a sight and sound extravaganza. Even before that first attendee walked through the door, the exhibitors with offerings in the "sound" area were tuning (or turning) up the band. At several points during the show, the sound level reached that of a rock concert. Fortunately, nobody suffered any permanent damage and most seemed to actually enjoy the competition.

MicroIllusions had one of the larger booths in the hall, exhibiting a variety of available titles. Demonstrations of Photon Paint were shown on a large screen projector so no one had any trouble getting a good view. On the other side of the exhibit, MicroIllusions had Music-X, a music software package that interfaces with MIDI devices. It was apparent that a great amount of effort has been put into making the software an excellent product. The system supports real-time recording, includes a number of editing features, and has a configurable librarian that is capable of handling almost any size sample.

Brown-Wagh Publishing's booth contained packages from most of the developers that they market, including Zuma Group, PAR Software, and the Softwood Company. Brown-Wagh just announced a new product they will be publishing called MIDI Magic, developed by Circum Design. MIDI Magic appears to be a promising product for the music novice, as well as, the music expert. The screens use windows, gadgets, and menus, much like other Amiga products currently available. If you can't remember what to do, just press the HELP key and the online help screen will be displayed. The program's controls are based upon the standard tape deck, with play, record, pause, rewind and fast forward. Other interesting features include the use of a "fuel gauge" for the amount of memory left and a metronome to count the beats. Beginners will probably have no difficulty using this package to delve into the world of MIDI.

Precision Incorporated's newest product is Pro*Sound Designer. It is a new sound sampler that will sample up to 32kHz in mono and 16kHz in stereo. With the software included in the package, it is possible to edit up to four sound samples at once. The user interface looked very good and seemed quite easy to use. Midi-Plus software included in the Pro*Sound Designer package allows sounds recorded with the sampler to be played back through a MIDI keyboard, or the Amiga. Pro*Sound Designer lists for $159.95.

Owners of Precision Incorporated products can now get technical assistance through the Official Superbase Information Network (OSIN) on American People/Link. According to a press release, Precision plans to implement a support program for developers wishing to develop and market their own Superbase applications. The OSAD (Official Superbase Application Developer) program will give developers access to proprietary information about Superbase, as well as marketing assistance for products developed with Superbase. This program should be in place after November 30.

Aegis astounded a great many attendees with its impressive list of audio and visual products including two or their newest titles, Lights! Camera! Action!, and AudioMaster II. Lights! Camera! Action! offers the ability to combine IFF pictures, IFF sounds, Aegis Sonix scores, and ANIM style animations into a single presentation. Aegis thinks this product supplies the necessary capabilities to develop business and educational presentations, as well as, videos for product demonstrations. AudioMaster II is a full blown digital sampling and editing package. It allows you to alter and store full stereo sounds into a true digital stereo sample. Sampling rates up to 44K are supported and if your Amiga is equipped with a 68020, AudioMaster II will support a rate of 56000 samples per second in mono or 52.6K in stereo. Of course the samples produced by AudioMaster II are usable with Lights! Camera! Action!.

The Business side of the Amiga

Business and productivity tools are proving that the Amiga is, indeed, a powerful machine. Anyone who is inclined to say the Amiga is just a great graphics machine has not seen what is available. Numerous vendors had exhibits to demonstrate their wares for the home and business.

One of the newest programs to enter the productivity world is MAGELLAN, from Emerald Intelligence. MAGELLAN is a software system designed to simplify the development of expert systems. Using this package, it is possible to capture information from human experts, which can then be used to help aide "non-experts". The system is built around "IF - THEN" rules or constructs, the same way that a great many "human" decisions are made. Using these rules and statements like, "If the system won't turn on then the electricity must be off", MAGELLAN makes it possible to build an expert system to help solve computer problems. In fact, it would be interesting to find out if they have thought about designing a technical support expert system using MAGELLAN.

Software Visions flew coast-to-coast to let visitors to AmiEXPO see their latest version of Microfiche Filer. Microfiche Filer Plus has all the same capabilities of its predecessor and has added a number of new features. Now included in the product is the ability to store HAM and overscan graphics images, automatic number formatting, automatic field calculations, and a full AREXX interface. The AREXX interface is especially useful because it allows the product to "communicate" with other AREXX compatible products. Using Microfiche Filer, it is possible to store thousands of records while being able to retrieve or sort them quickly and easily. The new HAM graphics support means that it is possible to build databases of high resolution pictures that could be used by "pictorial" based businesses like real estate or product marketing.

For those requiring modem communications, Oxxi Incorporated now offers A-Talk III. This package was developed by Felsina Software and includes all the standard features that everyone expects from communications software, plus a few more. It offers specific support for 10 different modems and a generic modem for custom configurations. A-Talk is also capable of emulating 7 different popular terminal types. It supports graphics exporting to such programs as Deluxe Paint and Aegis Draw. If you don't like to type or can't always seem to remember a complicated sequence required to log into a bulletin board then the script language will help you out. A "learn" mode is offered that will build the script from recording the steps you perform. Once you have a script, it is then possible to edit it to make specific changes. Finally, A-Talk III has also joined those offering AREXX support. As an example of how useful this feature is suppose you don't have a client's number in your phone list in A-Talk III. Using the AREXX interface it is possible to query a database product (like Microfiche Filer Plus) to retrieve the number and return it to A-Talk III, which can then place your call.

Oxxi also offers several other packages for the home and business use. MaxiPlan 500 and MaxiPlan Plus are high powered spreadsheet programs offering traditional functionality while taking full advantage of the Amiga's unique environment. Nimbus 1, a small business accounting package, is offered for those who don't need or want overly complicated features. With Nimbus all the accounting functions are running concurrently, which means that in the middle of updating an account you could write a check to someone. It is good to see that vendors recognize and use the power of a multi-tasking machine like the Amiga.

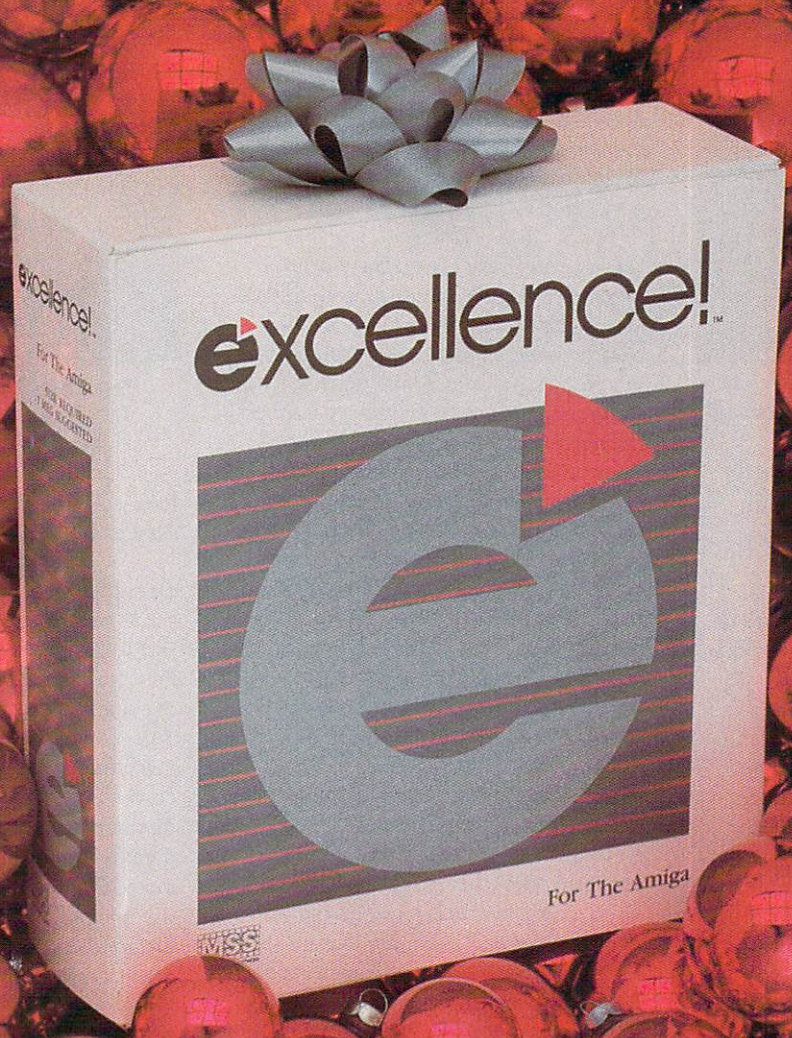
Micro-Systems Software received a lot of attention by demonstrating the usefulness of its packages Excellence! and The Works! Platinum Edition. Excellence! is a WYSIWYG word processor that allows graphic images to be placed in the document and has a "spell check as you type" feature. The Works! is five programs integrated into one. Included is word processing, telecommunications, a database, a spreadsheet, and a sideways printing utility. With The Works!, Micro-Systems has tried to address the most common needs of the home or small business user.

Micro-Systems Software attracted a lot of attention to themselves by using an "actor" inside the booth. A Madonna look-a-like lip-synced several songs, drawing people from all parts of the hall. Although, it may have been a little overdone, it did work. At a get together arranged by MSS on Saturday night, another person that closely resembled Elvira was out on the dance floor drawing looks from everyone.

WordPerfect Corporation was present demonstrating the power and ease of use of its word processor. WordPerfect offers a new product called the Library for the Amiga. The WordPerfect Library includes a Calendar to help keep track of important dates and appointments; a File Manager to help organize and manage a personal database; a Calculator so you won't have to dig around on your desk looking for your real one; a Notebook to help maintain lists; and finally a Program Editor. This is a powerful text editor which has left out those word processing features that aren't useful to writing programs, while including programming features that aren't useful to word processors.

ProScript is the latest product offered from New Horizons Software. This software package now makes it possible to print their ProWrite word processor files on a Postscript printer. If you don't have a Postscript compatible printer, ProScript can output the document to disk and you can send that file to someone that does, like a professional printing service. Having access to the

INTELLIGENCE IS A GIFT



Don't limit your potential! Experience excellence!, a wordprocessor designed for your Amiga, with 250 available fonts, a Spell-As-You-Type 90,000+ word Dictionary, Grammatical/Style Checker, Thesaurus, Index and Table of Contents generator, Headers, Footers and Footnotes! Skate through PostScript output, True WYSIWYG, automatic Hyphenation, Math, beautiful resizable Color Graphics, flexible Mail Merge, Columns and an easy-to-use Macro-Language making complex actions more fun than a sleigh ride! The fastest wordprocessor for your Amiga is the only one you'll ever need! Have an excellence! holiday!

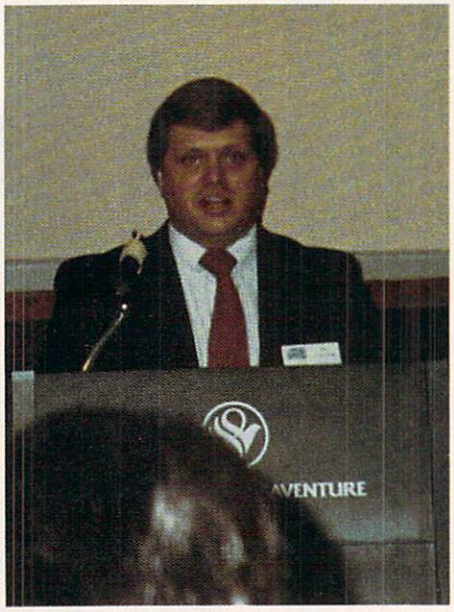
MSS
Micro-Systems Software

Committed to excellence since 1978

12798 Forest Hill Boulevard ● West Palm Beach, Florida 33414 ● 407-790-0770

See your local dealer or call our Sales Division 1-800-327-8724.

Amiga is a registered trademark of Commodore Business Machines ● PostScript is a registered trademark of Adobe Systems, Inc.
excellence! is a registered trademark of Micro-Systems Software, Inc.



Don Hicks from Amazing Computing asked all Amiga users to get involved!

Postscript world means that you will be able to produce "near" typeset quality documents without the need of a typesetter.

Soft-Logik Publishing Corporation has gone a step further than most by offering a complete desktop publishing product called Publishing Partner Professional. This package offers the features of a full blown word processor as well as many of the features required for typesetting, including Postscript output. If you don't have a Postscript printer, it offers full dot matrix support, including color printers. The program also includes a number of graphics import capabilities and has its own art program too, allowing you to include art inside you documents and flow text around it.

Of course few of the great products available on the Amiga are possible without program languages. Lattice, Inc. has announced that Lattice C 5.0 should be shipping in November. We were impressed with the new features and improved capabilities of Lattice C. Building on the many features that have been offered in past versions, 5.0 will be the new state-of-the-art C programming language for the Amiga. It will come with two binders of documentation that appears to be well written and organized. One of the new additions to the product is a debugger that will allow breakpoints to be set in multiple task simultaneously. If you have ever tried to debug a program that communicates with another, you can appreciate this feature. Several other programs and utilities will be included with the product to aide in program development. If you are considering purchasing a C language or upgrading your existing one then check into Lattice.

Naturally a great many games were on display at AmiEXPO. Some we have already mentioned but many we have not. Games attracted a lot of attention at the show, which just goes to prove that Amiga users also know how to have a good time.

Haitex Resources stayed busy from beginning to end demonstrating their X-Specs 3D glasses. These enable the wearer to experience a new dimension in computer graphics and game software. Programs have to be written to support the glasses which contain high-speed liquid crystal shutters. Opening and closing the shutters independently at 30 frames per second causes each eye to see a slightly different version of the screen. This is similar to the technology used to make 3D movies, except they use filters to alter what each eye sees. Looking at the screen without the glasses is much the same as looking at a 3D movie without their glasses.

Another interesting game just out is called StellarX from Laser Gamesmanship. This game might remind you of the old Asteroids arcade game, but it is much more than that game ever was. There are dozens of levels to traverse, more aliens to conquer and terrific arcade type action in this game. I wish I had a quarter for everyone that tried this game at AmiEXPO. A unique feature of the game is that is has online help, which is available by pressing the Help key.

Microdeal had several interesting titles on display at their booth. Their latest game is called Tanglewood. Tanglewood is a graphics adventure game, but no text entry is required. The basic story revolves around your search for some very important documents. It's not as easy as you might think — because you are on an alien planet with a harsh environment and hostile competitors. You have 5 old style "mobiles" at your disposal but they each have different abilities. To solve the game you will

Perry Kivolowitz received the first Amazing Computing Amiga Achievement Award

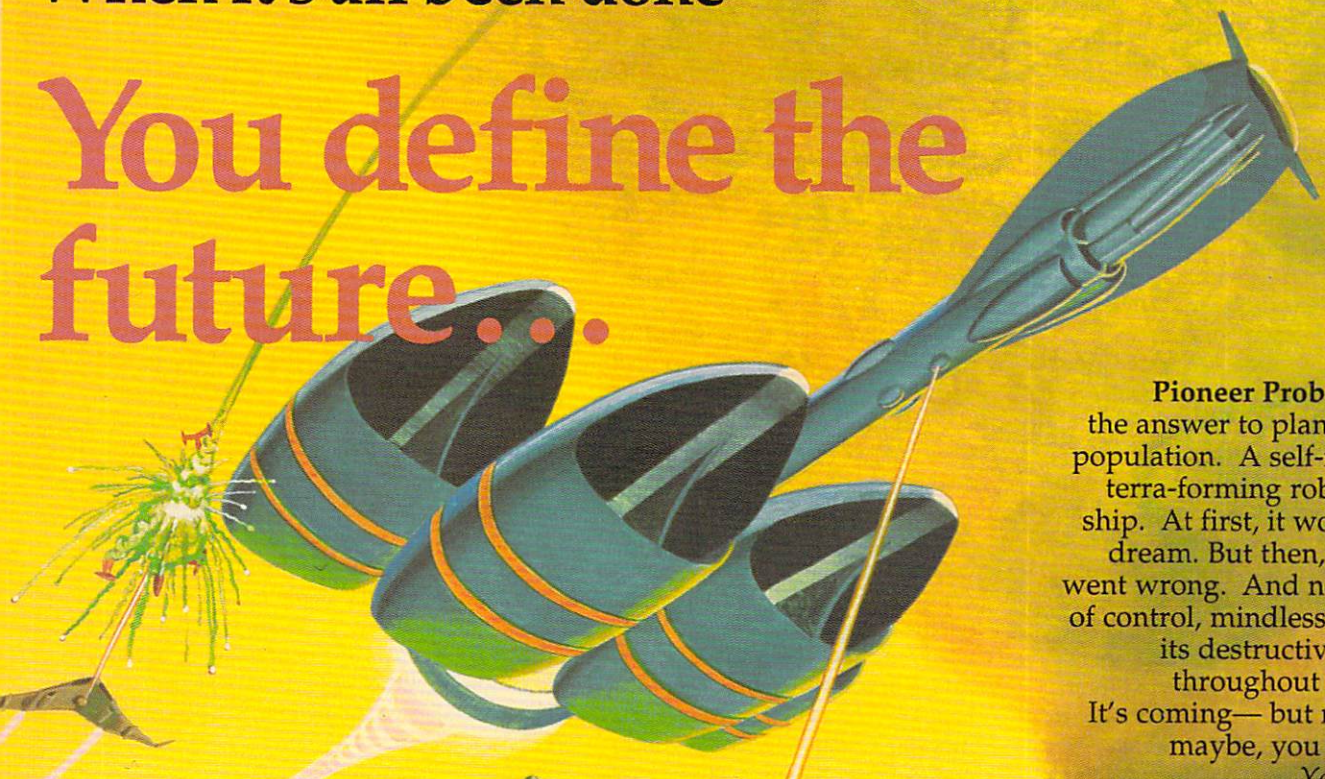
have to learn which to use in certain situations.

One title that turned a lot of heads at the show was in the Free Spirit booth. Hardly anyone could resist stopping (at least for an instance) to look at the package for Sex Vixens From Space. The only thing that we'll tell you about this game is the disclaimer that appears on the box: "Free Spirit Software, Inc. disclaims any responsibility for alleged damages, consequential or inconsequential, resulting from inappropriate use of this game. This includes, but is not limited to, allegations of eyestrain, near-sightedness, high blood pressure, feeble-mindedness, priapism, nymphomania, nervousness, insanity (temporary or permanent), unwanted hair growth or loss, mental incapacitation, excessive perspiration or general physical debilitation." What more could we say?

Starvision International introduced three new entertainment games that are to be released this year. Mega Pinball is an arcade style pinball machine game boasting hyper sound effects and superb graphics. Twin Ranger is another arcade style game that can handle 2 players while smoothly scrolling horizontally and vertically. Snowberry is a game based upon a weekly televised program. In this game you control Snowberry, a little bear, by making him jump from one ice

When it's all been done —

You define the future...



Pioneer Probe Mark IV: the answer to planetary overpopulation. A self-replicating, terra-forming robotic spaceship. At first, it worked like a dream. But then, something went wrong. And now, it's out of control, mindlessly spewing its destructive offspring throughout the galaxy. It's coming— but maybe, just maybe, you can stop it. *You must try!*

The first game with "HAM Mode" Graphics for the Amiga

BROUGHT TO YOU BY

TERRIFIC™ SOFTWARE

Look for Pioneer Plague, only $39.95, at your favorite computer store, or call 800-234-7001

Pioneer Plague:

Defines the future in Amiga game development:

- Executed in "HAM mode", bringing 4,096 colors to the screen.
- Designed to instantly challenge you with the excitement of its arcade-style action while drawing you into an adventure set against a vast galactic tapestry.
- Sub-Euclidian hyper travel
- Orca Drones: programmable combat support aircraft. You design your own offensive and defensive tactics. Every time you play, it's different!
- A sinister cast of increasingly clever opponents.
- Qualitative scoring to help you improve your game play.
- Original stereo music score
- Digitized sound effects

PIONEER PLAGUE™

by Bill Williams

block to another without falling in the icy water.

There were many more exhibitors and products at the show than could be covered in this article. These we have mentioned were just a few of the most noticeable at the California AmiEXPO. Of course, there's more to "The Amiga Event" than products and booths. Several keynote addresses were made by prominent people from the Amiga industry and seminars were held to exchange information about program development on the Amiga. Joel Shusterman, vice president of Marketing for Commodore, gave the first keynote address. A large audience was on hand to hear from the former president and founder of the Franklin Computer Corporation. (Franklin made Apple compatible equipment). He expressed great enthusiasm at the prospect of being able to market the Amiga, calling it "the best kept secret in the industry". He hopes to turn around the Amiga market, and get the word out to the rest of the world.

Shusterman said that Commodore will begin focusing on the professional video, home video, graphic arts, and multi-media markets. A videotape aimed at the graphic arts market was shown during the speech. Mr. Shusterman indicated that, in the near future, Commodore would be making more video tapes aimed at other markets.

According to Shusterman, within the next ninety days, the following products will be shipping from Commodore:

- o A2620 - The 68020 board.
- o A2286 - The AT bridge board (XT and AT compatibility), with 1 MB RAM.
- o Amiga 2000HD - A bundled package that includes an Amiga 2000, a 40 MB/20 ms hard drive, and the 2090A hard disk controller.
- o Amiga 2500 - A bundled package consisting of the Amiga 2000HD and the A2620 card.

Hard disks that are shipped will already be formatted and configured, meaning that purchasers will be able to use them right out of the box. Mr. Shusterman also stated that there will be an upgrade program for owners of the 2090 hard drive controller to the newer 2090A card.

Don Hicks, managing editor of Amazing Computing gave the second keynote. Mr. Hicks talked about a variety of different things concerning Amazing Computing magazine and the Amiga community.

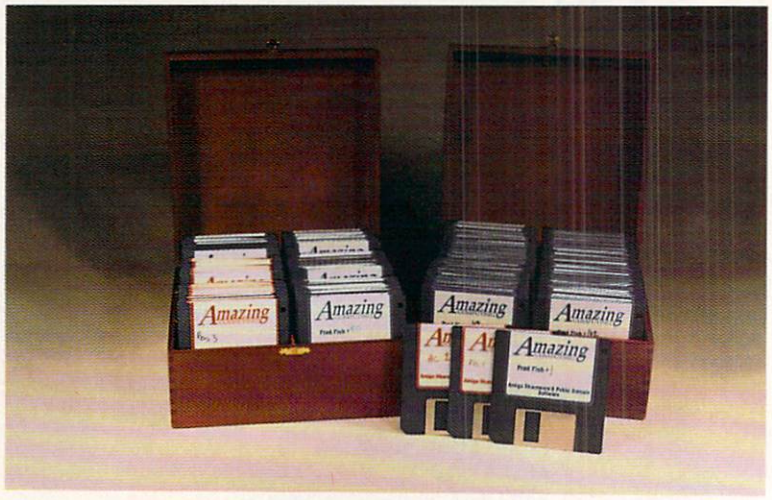
Mr. Hicks applauded all the developers and Amiga owners who are helping to make the Amiga a success. He stated that two years ago, there were 186 developers in the Amazing Computing product guide. As of this writing, the product guide that is currently being compiled will have over 600 developers, and almost 1200 products for the Amiga.

Continuing his discussion about the magazine and the Amiga, Don suggested every Amiga user had a responsibility to tell the Amiga developers and vendors what they expected and needed in Amiga products. He suggested, "If you have any comments or suggestions for Amazing Computing, or any Amiga company, send them a letter! Letters receive a tremendous amount of respect. Remember, you are the best resource we Amiga developers have."

Don Hicks had the honor of announcing the winner of the first Amiga Community Service Award. This award was given to Perry Kivolowitz, president of ASDG. Don spoke of Perry's contributions, not only through ASDG, but also throughout the Amiga community in both hardware and software. Perry, who co-authored the Amiga Working groups proposal, is working to bring developers in the Amiga community together to help enhance the Amiga. Congratulations Perry!

Fred Fish gave the keynote address on the last day of AmiEXPO. A sizeable audience gathered that morning to listen to the famous software "packrat" (as he called himself) of freely redistributable software. In his address, he explained how the library got started and how he decides what software will appear on the Fred Fish disks. He also addressed what the terms "public domain" and "freely redistributable" really mean and how important such software is to the Amiga. Mr. Fish also spoke of what he hopes the future of the Amiga will hold and ended his talk by fielding questions from the audience.

Of course Amiga shows are more than just products and speeches — they're fun too. It's a time and place where Amiga users can get together to rub elbows with each other and have a good time. In fact, they are almost like family reunions! The next time AmiEXPO is in your part of the country, try to take a day and visit. You won't be disappointed.



Congratulations!

Three Amiga users won a complete set of Public Domain Software from Amazing Computing

Chris Lavoire
Carson, CA
Ron Wilson
Garden Grove, CA

Penny M. Karlberg
Long Beach, CA
A special thank you to all those individuals who participated.

MASTERPIECE PROFESSIONAL FONT COLLECTION©

20 DISK SET

The largest collection of fonts and clip art available in a single package for the AMIGA.

110 DIFFERENT FONT STYLES

This doesn't mean 10 sizes of 11 fonts. It means 110 DIFFERENT fonts.

LARGE SIZES

Specially designed for video work. 95 % of the fonts are over 100 pt. tall. Easily resized smaller.

PATTERN CLIP ART

141 hi-res DPaint II pages. There are thousands of objects and examples.

ALL FONTS ARE HI-RES

BRUSHES - 2 disks full of color brushes.

COLORFONTS - 4 full disks.

100 PAGE MANUAL - Full size font printouts.

20 DISK SET - ONLY $199.00

Contact your local AMIGA dealer or order direct from AROCK Computer Software, 1306 E. Sunshine, Springfield, MO 65804 1-800-288-AROK

AmiEXPO California '88 Exhibitors

A-Squared Distributions, Inc.
6114 LaSalle Ave.
Suite 326
Oakland, CA 94611
(415) 339-0339

Aegis
2115 Pico Blvd
Santa Monica, CA 09405
1(800) 458-5078

Amazing Computing
P.O. Box 869
Fall River, MA 02720
(508) 678-4200

Amiga Sentry
706 Old Marple Rd
Springfield, PA 19064
(215) 544-6394

AmigaWorld
80 Elm Street
Peterborough, NH 03458
(603) 924-9471

AmigoTimes
5124 St. Laurent
Suite 100
Ville Ste. Catherine
Quebece JOL 1EO Canada
(514) 638-6303

AMnews Corporation
P.O. Box 1389
Guerneville, CA 95446-1389
(707) 887-9708

AMuse, The New York Amiga Users Group
151 1st Avenue, Suite 182
New York, NY
(212) 460-8067

Antic Publishing
544 2nd St.
San Francisco, CA 94107
(415) 957-0886

ASCIITECH/Augnet
141 Del Medio Ave, Ste. 210
Mt. View, CA 94040
(408) 742-8817

ASDG, Inc.
925 Stewart Street
Madison, WI 53713
(608) 273-6585

Audio Transcripts
610 Madison Street
Alexandria, VA 22314
1(800) 338-2111

B-Side Design
701 Sophia Street
Fredricksburg, VA
(703) 371-4455

Brown-Wagh Publishing
16795 Lark Ave
Suite 210
Los Gatos, CA 050300
1(800) 451-0900

Byte by Byte Corporation
3736 Bee Cave Rd.
Suite 3
Austin, TX 78746
(512) 328-2985

Commodore Amiga User International
40 Bowling Green Lane
London, United Kingdom
011-441-278-0333

Commodore Business Machines, Inc.
1200 Wilson Drive
Whest Chester, PA 19380
(215) 431—9100

Compu-Art
P.O. Box 712, Victoria Station
Montreal, Quebec H3Z 2V8 Canada
514-483-2080

Comp-U-Save
414 Maple Ave
Westbury, NY 11590
(516) 997-6707

COMPUTE!
324 W. Wendover Ave
Suite 200
Greensboro, NC 27408
(919) 275-9809

Creative Computers
4453 Redondo Beach Boulevard
Laundale, CA 90260
(213) 542—2292

Creative Microsystems, Inc.
10110 S.W. Nimbus
Suite B1
Portland, OR 97223
(503) 620-3821

Digital Creations
530 Bercut
Suite F
Sacramento, CA 95827
(916) 446-0270

Digital Dynamics
739 Navy St.
Santa Monica, CA 90405
(213) 396-9771

Elan Design
P.O. Box 31725
San Francisco, CA 94131
(415) 621-8673

Emerald Intelligence
334 South State Street
Ann Arbor, Michigan 48104
(313) 663-8757

EyeBytes
1900 Brooklane, E-9
Ellensburg, OR 98926
(509) 962-6570

Finally Technologies
1377 9th Avenue
San Francisco, CA 94122
(415) 564-5903

Free Spirit Software, Inc.
58 Noblec Street
Kutztown, PA 19530
(215) 683-5609

Fuller Computer Systems
706 W. Broadway
Suite 202
Glendale, CA 91204
(818) 243-4540

Gold Disk
P.O. Box 789, Streetsville
Mississauga, Ontario
Canada L5M 2C2
1(800) 387-8192

Great Valley Products
P.O. Box 391
Malvern, PA 19355
1(800) 426-8957

Haitex Resources
208 Carrollton Park
Suite 1207
Carrollton, TX 75006
(214)) 241-8030

Innovision Technology
P.O. Box 743
Hayward, CA 94543
(415) 538-8355

Interactive Video Systems
15201 Santa Gertrudes Ave.
Suite Y 102
La Miranda, CA 90638
(714) 994-4443

Jumpdisk
1493 Mt. View Ave
Chico, CA 95926
(916) 343-7658

Lattice Inc.
2500 South Highland Ave.
Lombard, Illinois 60148
1(800) 533-3577

MAGNI
Magni Systems, Inc.
9500 SW Gemini Drive
Beaverton, OR 97005
1(800) 237-5964

Manx Software
1 Industrial Way
Eatontown, NJ 07724
(201) 542-2750

Memory And Storage Technology
Technical Excellence
7631 East Greenway Rd
Scottsdale, Arizona 85260
(602) 483-6359

Microdeal
576 S. Telegraph
Pontiac, MI 48053
(313) 334-8729

MicroIllusions
17408 Chatsworth St.
Granada Hills, CA 91344
1(800) 522-2041

MicroTimes
1800 N. Highland Street
Suite 220
Hollywood, CA 90028
(213) 467-7878

Micro-Systems Software, Inc.
4301-18 Oak Circle
Boca Raton, FL 33431
1(800) 327-8724

Mindware International
230 Bayview Drive
Suite 1
Barrie, Ontario L4N 4Y8 Canada
(705) 737-5998

Moniterm Corporation
5140 Green Circle Drive
Minnetonka, MN 55343
(612) 935-4151

New Horizons Software
P.O. Box 43167
Austin, Texas 78745
(512) 328-6650

NewTek, Inc.
115 W. Crane St.
Topeka, KS 66603
1(800) 843-8934

Oxxi Incorporated
P.O. Box 90309
Long Beach, CA 90809
(213) 427-1227

Precision Incorporated
8404 Sterling St.
Suite A
Irving, TX 75063
(214) 929-4888

RoboCity News
#3 Lagoon Dr.
Suite 180
Redwood City, CA 94065
(415) 593-2802

Ronin/Hurricane
P.O. Box 1093
Alameda, CA 94501
(415) 769-9325

Sedona Software
11828 Rancho Bernardo Road
#123-200
San Diego, CA 92128
(619) 451-0151

Silent Software, Inc.
706 W. Broadway
Suite 202
Glendale, CA 91204
(818) 243-4540

Soft-Logik Publishing Corporation
11131 South Towne Square
Suite F
St. Louis, MO 63123
(314) 894-8608

Software Visions, Inc.
P.O. Box 3319
Framingham, MA 01701
1(800) 527-7014

Spirit Tecnology
220 West 2950 South
Salt Lake City, Utah 84115
1(800) 433-7572

Starvision International
305 Madison Ave
Suite 411
New York, NY 10165
(212) 867-4486

Supra Corporation
1133 Commercial Way
Albany, OR 97321
(503) 967-9075

TritTec Corporation International
9276 Adelphi Road
Suite 102
Adelphi, MD 20783-2029
1(800) 284-3624

WordPerfect Corporation
288 West Center St.
Orem, Utah 84057
(801) 227-4010

Your Amiga
A.S.P.
One Golden Square
London, W1R 3AB United Kingdom
011-441-437-0626

BUG BYTES

by John Steiner

The time has finally come to announce a long awaited upgrade notice. If you have not been living in a vacuum, you may have already heard about the availability of **Workbench 1.3**. For those of you who have not yet received the upgrade, here are the details.

Workbench 1.3 has a suggested retail price of $29.95 and comes with both Kickstart and Workbench 1.3 disks plus documentation. Amiga 2000 and 500 owners do not, of course, need the kickstart disk, as your Kickstart is internal to the computer in ROM. This brings up a question of the necessity of upgrading the Kickstart ROMs. The only feature that the new Kickstart ROMs have that is not in the original 1.2 Kickstart ROM is a hard disk auto-booting feature. If you wish to have your Amiga 2000 or 500 boot directly from a hard disk, you will have to order a new Kickstart ROM. The ROM upgrade has a suggested retail price of $45.00.

The availability of Workbench 1.3 is expected to be in short supply. The upgrades may be purchased at your local Amiga dealer, if he has them. One local dealership was told when the upgrades were ordered, that all dealers would be receiving only 40% of the total number ordered with the first shipment. The remainder of the order would be filled as production catches up with demand.

Two other upgraded products have been released by Commodore. The A2090A auto-booting (with 1.3 kickstart) controller has replaced the A2090 which has been discontinued. The pricing remains unchanged. The A2052 RAM expansion board with 2 MB of RAM has been replaced by the A2058 board. The latter board has 8 MB of sockets, with 2 MB chips included. Pricing on this board is expected to be slightly higher than the older board, however, the current price was not available as of this writing. Also, for current owners of this hardware, there was no announced upgrade path, and none was expected.

According to a notice posted to PeopleLink, the HP PaintJet driver in **WordPerfect** needs a slight modification to make the bold function shut off. The command passed to the printer from the 8/10/88 version of WordPerfect is <27>(s)B. It should be <27>(s0B. The posting recommends correcting the mistyped character with WordPerfect's PrintDef program.

Early shipments of Gold Disk's unique publishing program **Comic Setter** have a problem with the printer drivers. If you have recently purchased a Comic Setter and it will not print a comic , you can return your disk to the dealer for replacement. After shipping several hundred of these packages (I was told by the folks at Gold Disk that the number of copies of the program that were pre-sold was tremendous), they found that they had accidentally shipped version 1.3 printer drivers with an incompatible version of printer.device. If you have a hard disk and have not tried to boot and print directly from your Comic Setter disks, you probably won't have noticed a problem. They were very apologetic that this problem occurred, and they promised that they were sending corrected disks to dealers directly for replacement. They also said that if a dealer could not help the customer with this problem, to contact Gold Disk directly. According to the representative from Gold Disk, dealers were shipped enough corrected disks to match the number of units originally ordered.

Gold Disk, Inc.
Box 789
Streetsville
Mississauga ONT Canada
L5M 2C2
1-800-387-8192

Aegis VideoTitler has a bug that involves operation from the WorkBench when used from a hard drive. It seems that VideoTitler only operates properly from the CLI when it is started from a hard drive. This same problem also exists with Aegis Draw Plus, as was reported in Bug Bytes in volume 3.1. Aegis technical support suggests starting both programs from the CLI rather than the Workbench.

One suggested workaround for those who really want to start their software from the Workbench is Xicon which is on the Fred Fish disks, and can now be found in the C directory of Workbench 1.3. Xicon allows you to execute CLI only batch files and programs by clicking on an icon. I haven't tried this fix with either of these programs, but it is worth the attempt.

Aegis Development
2210 Wilshire Blvd Suite 277
Santa Monica, CA 90403
(800) 345-9871

The exploding crop of computer viruses may be an annoyance for computer owners, but they are a real headache for software manufacturers. Developers need to be especially aware of their responsibilities in making sure their master disks don't become infected by one of the numerous viruses that are floating around the computing community. It would be easy to spread the virus to hundreds, even thousands of

(continued)

previously uninfected computers with just one mistake. Unfortunately this problem has occured on several occasions.

I almost hesitate to single out any manufacturer by pointing out that their originals might be infected, but a spokesperson for Sound Quest, manufacturer of Texture, provided a statement regarding the release of an upgrade to this high quality music oriented software package which I felt presented a clear view of the problems encountered by any software manufacturer in this age of viruses.

Unfortunately, in our haste to release **The Quest I: Texture**, there were several disks shipped with two versions of song files. The song files NOT in the Song Drawer are the proper ones. In addition, without our knowledge the SCA Virus cropped up. Needless to say, this did not improve our day. It can be removed by running the Install program, using a clean write-protected Workbench on the Quest disk. This will not damage the program. We have learned a real lesson and apologize to those inconvenienced. We believe in providing software support to our customers. And, anyone still concerned with their disk should call us with their warranty number and we will replace their disk happily. This is our 12th software product for the Amiga and we hope to provide many more in the future.

What we did to The Quest I: Texture was to:

1) Remove the need for the Roland MPU-401 hardware interface.
2) Maintain its ROCK-SOLID timing.
3) Provide a comprehensive pull-down menu/mouse/keyboard screen display.
4) Reduce the list price of Texture from $700 to $150.

We can be reached at:

Sound Quest, Inc.
5 Glenaden Avenue East
Toronto, Canada M8Y 2L2
416-234-0347

While on the topic of viruses, a program that everybody should have in their startup-sequence, **VirusX**, has been upgraded to version 2. VirusX, once executed, remains active and unnoticed until you put an infected disk in the drive, whereupon it notifies you of that fact. VirusX is public domain, and available on information services and bulletin boards everywhere as well as the Fred Fish collection.

3-Demon is a graphics program from Mimetics that has just recently been upgraded, adding several new features including the ability to save Turbo Silver 2.0 files, VideoScape Binary files, and Wavefront files. Also, bug fixes and improvements have been made to the package. Previous owners of the program can get an upgrade from Mimetics for just seven dollars to cover postage and handling. Send them your program disk and a check or money order for $7.

Mimetics Corp.
Box 1560
Cupertino, CA 95015
(408) 741-0117

Harry Evangelou has recently been hired as **Haitex's X-CAD** product manager. Harry has recently completed two add-on programs for use with X-CAD. A HPGL to X-CAD converter allows users to import their symbol databases from other packages until the DXF reader arrives, and an IFF brush to screenmenu converter that allows users to use any of the IFF drawing packages, such as Deluxe Paint to design custom menus. Harry has also created a custom menu template with several highly productive features.

These files have been made available to X-Cad users and can be found on Bix, Compuserve, PeopleLink, and other BBS systems by now. If you do not have access to any of the networks, you may receive the programs by sending a blank disk with a SASE (make sure you include enough postage) to the address below.

Harry Evangelou
X-CAD Product Manager
Haitex Resources
208 Carrollton Park - Suite 1207
Carrollton, Texas 75006
(214) 241-8030

Last month, I reported that Impulse, Inc. is now shipping **Turbo Silver** version 3.0 in both "Integer" and "Fast Floating Point" versions. At that time, complete details on the upgrade policy was not available. The upgrade to version 3.0, for owners of Turbo 2.0, is only $5.00 including a completely rewritten 150 page manual punched to fit into your Silver 3-ring binder.

If you are a registered owner of Turbo Silver, you should be receiving an upgrade letter from Impulse directly. If you haven't received a notice of the upgrade by the time you read this, call them about upgrading.

Impulse!
6879 Shingle Creek Pkwy Suite 112
Minneapolis, MN 55430

According to the Gallery 3-D Newsletter from Byte-by-Byte, there will be an upgrade policy for those who own Sculpt-3D and/or Animate 3D to **Sculpt-Animate 4D**. The new program will have a list price of $499. If you already own Sculpt 3D and Animate 3D and have sent your warranty cards in, the upgrade price will be $195.

There are many new features to the program, including an enhanced user interface, grids and grid snap, user definable keystroke macros, support for 68020/30 and 68881/82, and many others.

Byte by Byte
9442 Capital of Texas Highway N.
Austin, TX 78759
(512) 343-4357

That's all for this month. If you have any workarounds or bugs to report, or if you know of any upgrades to commercial software, you may notify me by writing to:

John Steiner
c/o Amazing Computing
Box 869
Fall River, MA 02722

...or leave EMail to
Publisher on People Link
or
73075,1735 on CompuServe

•AC•

MAGELLAN

The Amiga Gets Smart

by Steve Gillmor

Remember the day you first turned on your Amiga and entered the multitasking world of computing. How about the thrill of DPaint 2, where you suddenly had a Paintbox on your desktop with features and effects rivaling those previously available only at two or three hundred dollars an hour. 3D animation and raytracing you can't buy at any price. Desktop video when desktop publishing was barely out of its diapers.

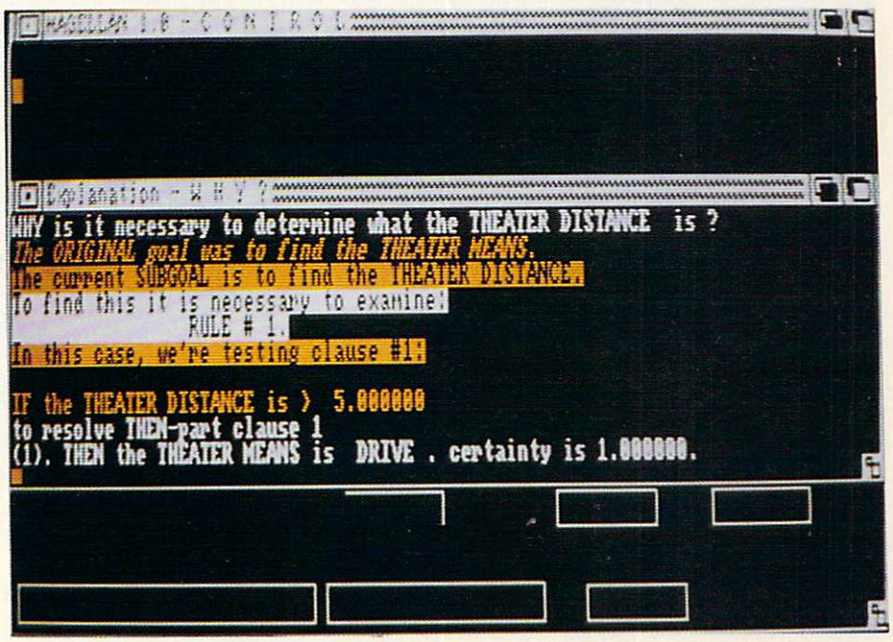
In every aspect of computing, the Amiga has ventured where no computer has gone before, pushing out at the boundaries of the state of the art while offering power at prices so low they brought in the first-time user in numbers now nearing one million.

Now comes the world of artifical intelligence in the form of MAGELLAN, expert system software for the Amiga from Emerald Intelligence. With the aid of a well-written manual, you can jump right in, but first let's backtrack with a little history.

The AI Story

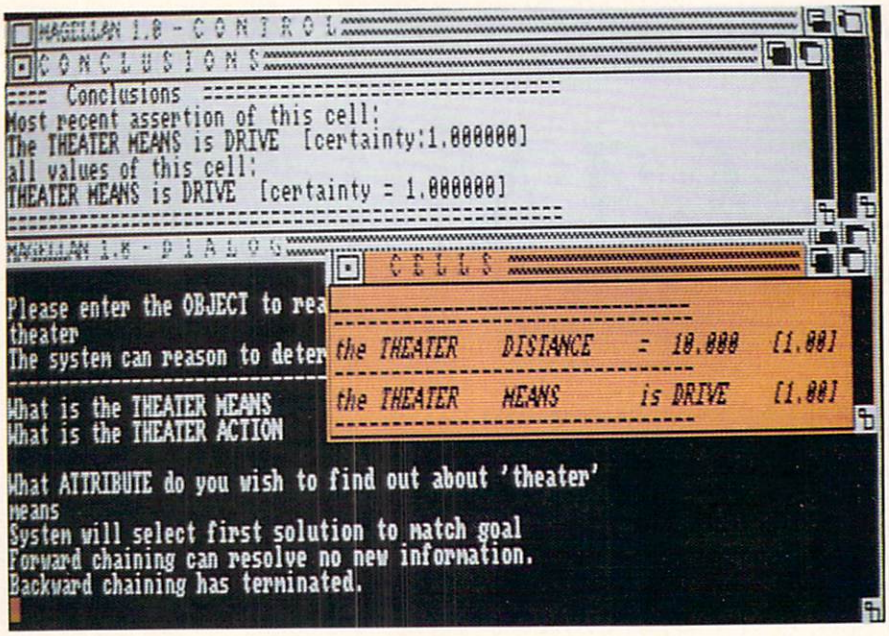
The story of AI is really the story of computing at an early crossroad. With the end of World War II, American and British scientists began applying their resources toward the development of what was to become the computer. Each team began with the same basic system: An electronic machine driven by stored program directions to carry out numerical calculations. The British wanted these instructions to be based on logical operators such as "and," "or," and "not." These operators could be used to assemble more specialized numerical operators for arithmetic calculations, and to manipulate symbolic material such as statements in ordinary language.

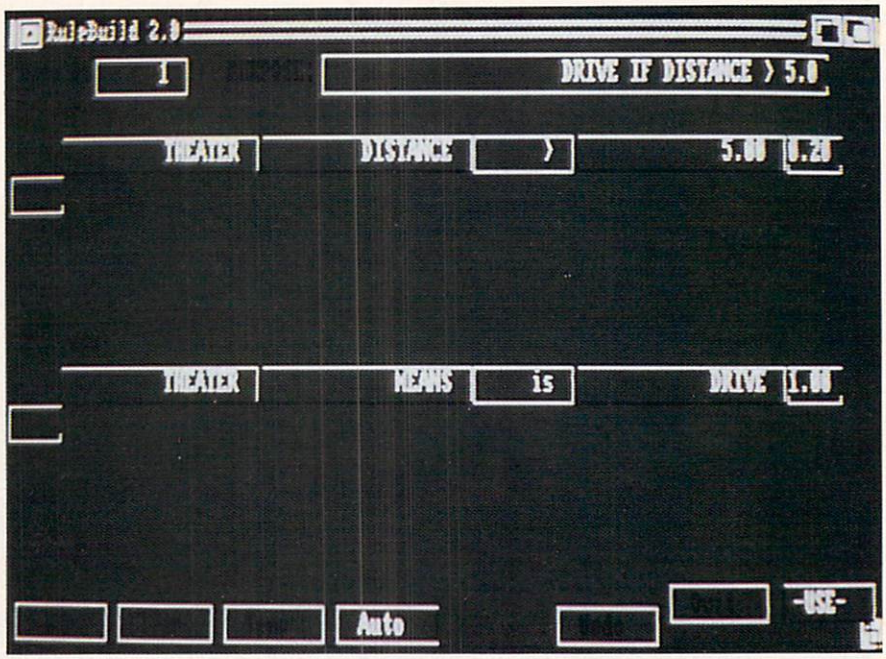
The Americans, however, were more interested in a simpler, faster machine to do arithmetic calculations, so they used numerical operators such as "+," "-," "<," etc. The British went along with this and confined AI work to a loose consortium of computer scientists and psychologists in theoretical research at the university level. In 1950 Alan Turing, leader of the British team, tested the theory of "machine intelligence" in a paper entitled "Computing Machinery and Intelligence." He posited placing a human in room A, an "intelligent" computer in Room B, and a second human "interrogator" in Room C. He or she communicates with Rooms A and B via teletype, and does not know who or what is in which room. If the interrogator cannot distinguish between the two responses then the computer in Room B is declared "as intelligent" as the person in Room A.



Why? explains why information is requested. The rule clause being investigated is displayed as well as the value sought to prove the rule.

(continued)





(Top) On opening MAGELLAN, the bottom of the screen is filled with the DIALOG window, which acts as an interactive message board. The top third of the screen is the CONTROL window, where help messages are displayed upon request.

(Bottom) Build rules in the Rulebuild window using a simple, English-like method.

That was the theory. Getting a computer smart enough to do that, or a human stupid enough, has occupied much of the past forty years. It is only now with the advent of high-speed powerful chips and sophisticated software that we are learning how to inculcate a machine with the properties of the "expertise" of the human in various disciplines. Most expert systems work by encapsulating the knowledge of the expert in a series of IF-THEN rules. These rules are made up of the stuff of what we call intelligence: facts and heuristics, or rules of thumb. This surface (or experiential) knowledge is distinguished from deep knowledge of formal principles and theories. Existing expert systems only provide good advice when they are used to assist users in solving problems that lie within narrowly defined domains. One good rule of thumb for MAGELLAN is that IF you can do something with your Amiga, THEN the chances are good you can do it under control of a rule base.

AI with MAGELLAN

Let's do just that: Fire up MAGELLAN and go through its paces under the "control" of its 132-page manual and two disks, the program on a bootable 1.2 Workbench, and a data disk full of sample knowledge bases. The manual is divided into three main sections: Level One, a Menu Options Overview; Level Two, Usage of the aforementioned Options; Level Three, Getting Started and a walkthrough tutorial on building a small expert system. A glossary that efficiently brings you up to speed in AI terminology follows, and an Appendix Introduction to AI runs the gamut from answering "What is Artificial Intelligence?" to "What applications has AI been successful with?"

The Components

An Introduction describes the three major components of expert systems:

1. **The Inference Engine** - the program that makes decisions, asks questions, and does the 'work'.

2. **The Interface** - all of the software and hardware that connects the Inference Engine to the outside world including input via menus, keyboard, mouse, sensors, etc., and output through screens, windows, printers and displays.

3. **The Knowledge Base** - what the expert system knows about. An expert system to diagnose an automobile would have a specialized KB containing info about carburators and alternators.

This separation of powers, if you will, is one of the secrets of AI. Ninety five percent of the time spent debugging computer software is writing, waiting for compilers to compile, linking, re-editing, compiling. With knowledge-based programming, you change a rule, click and enter it, and that's it. MAGELLAN lets you break down a problem into simple IF-THEN rules that you enter in simple sentences like: IF car is not starting AND weather is rainy, THEN distributor cap is cracked. As few as 10 or 20 rules can form a useful system. But unlike software where we may never see DeluxePaint 3, we can change and add to the rules of a knowledge base to respond to new information as we get it. The Inference Engine remains intact, but the separate Knowledge Base and a variety of inputs changes.

You're in CONTROL

When you load MAGELLAN you come upon two windows set against the Emerald Intelligence green color 0 background. The bottom of the screen is filled with the DIALOG window, which acts as an interactive message board. The top third of the screen is the CONTROL window, where help messages are displayed upon request. Clicking in the CONTROL window with the left button allows you to activate the Menu Options with the right button. There are seven general categories as you move left to right: Session, Rules, Cells, Words, Inference, Display, and Explanation.

Session is just another word for Project, and its choices are basically familiar functions with the vernacular of expert systems. Knowledge bases are composed of rules; rules are made up of object, attribute, and value. For example, IF fur (object) color (attribute) is brown (value), THEN dog (object) type (attribute) is watchdog (value).

So the Session sub-menu gives you the choice of Loading a complete Knowledge Base, Loading or Saving specific sets of values, Clearing all inputed values, Help, Quit, or Status Display. The latter displays current program and system information including the current directory and existing knowledgebase files as well as the number of rules, words, and cells in the current KB. This command offers immediate gratification at any moment, something you will appreciate as you fumble your way toward Intelligence, Artificial and otherwise.

Commands added since printing the manual include Lock, a feature for use in testing different combinations of rules and values, and New CLI, which I am using right now to multitask with Scribble! as I write this article. Also added since the manual went to press are optional keyboard commands for most menu items.

Rule Options include: Select, Create, Display, Save Rules, Edit, Delete, and Help. You can enter new rules with Create, or change existing ones by Selecting, then Editing or Deleting. Saving Rules allows you to store tested rules in a separate file and "back up" a current rule base while experimental changes are made. Help, as with all such menu choices, directs you to the appropriate pages of the manual.

Cell Options are Select, Display, and Help. Each rule created, automatically creates a cell. A cell is one of a number of combinations of the object, attribute, and value (OAV triplet) in a rule. We'll get back to cells in a minute, but they are important because it is by Selecting cells that you can attach variables, text files, images, or programmed functions to an individual cell to expand the complexity and intuitive reasoning of your expert system.

Words Options allow you to Add Words, Display the dictionary list of word and phrase entries, create Synonyms, and Edit mispelled words or change them to another word. Such changes are automatically global in the rule base. Help is a Synonym for Pages 59-64 in the manual.

It's Logical

Inference is defined in the glossary as "The process by which new facts are derived from established facts." Our Menu choices in the manual are Backchaining, Forward Chaining, and Synergistic. MAGELLAN tries to derive the information it needs from the rules already in the system before it questions you for input. Let's recall that rule about the dog. Backchaining is initiated by a goal. The system attempts to determine a Value for the goal and identifies rules that conclude with a Value for the goal. If the goal is "watchdog", then the rule that pertains is our "fur type is brown, THEN dog type is...." So having located that rule, the process then Backs up and attempts to determine if the IF clauses of the related rules are true by determining

(continued)

values based on yet other rules. Eventually the backchaining sequence arrives at either a pause as it asks the user for information it doesn't have, or it may halt with the result stated as unknown. Successful backchaining can result in an answer to the original goal; if more than one solution is possible, a list of acceptable solutions is displayed.

Forward Chaining begins by reviewing the known facts and then "firing" all the rules whose IF clauses are true. The system then begins another cycle, checking what additional rules may now be true, and so on until the program reaches a goal or runs out of new possibilities. Synergistic inference uses both backward and forward chaining, backchaining to resolve a goal, then moving forward to locate all associated conclusions and triggering related processes like IFF and text files or external program commands. Synergistic is the default mode, since its result reflects the most comprehensive up-to-date analysis of information available to the system.

Attempt and Goals

Scrolling down the Inference menu, you come to several important additions to the program, Attempt and Goals, documented in MAGELLAN Manual Supplement 1.0 included in the package. With Attempt, the goals you initiated Backchaining with previously can now be created, once prioritized and stored with a knowledgebase. As the knowledgebase is retrieved with the Load KB command from the Session menu, the saved goals are restored. Goals Option has a sub-menu with choices to Add, Delete, Display and Clear. By selecting Attempt you initiate Backchaining in decreasing order of priority from 100 (high) to 0 (low). All in all, a neat time saver that encourages experimentation and automates the user interface for those of us like me who appreciate digital hand-holding.

Display Options

Display Options include Preview Picture, Output Mode, and Results Mode (the latter yet another recent addition). Preview lets you search for graphics or pictures without leaving MAGELLAN. Output can be directed via a sub-menu to Screen, Printer, or File. A Voice setting is grayed out, indicating a "hook" to be activated in revisions forthcoming soon. Default is to the screen, of course, but you can direct to multiple destinations simultaneously. Occasionally, it is desirable to suppress the display of resolved information and goals during backchaining. Results Mode allows this, and via the Wait sub-menu command, a time delay between results windows.

Let me Explain

Our last Menu Option is Explanation, and not a minute too soon. But seriously, these options show the path of reasoning used to conclude the current result. Why? explains why information is requested. The rule clause being investigated is displayed as well as the value sought to prove the rule. Trace shows the path of reasoning that was followed to get to a particular result. This is especially helpful for the beginning Knowledge Engineer, as it shows how the system came to its conclusions and where the rule base took any wrong turns. As the manual points out, expert system development programs are called shell programs because they themselves contain no knowledge about a problem, but instead enable you to create your own expert systems rapidly. You - the expert - teach the computer how to solve the problem by entering rules explaining the steps involved in the decision-making process. Why? and Trace are like little windows into MAGELLAN's brain; if MAGELLAN asks the wrong question or returns an unexpected result, these options let us track down where we went wrong and help to correct our reasoning.

Creating a Rule

Now that we've checked out the Menus, MAGELLAN is ready to go. So are we, once we've learned a few more terms as we explore Level Two. The manual navigates back through the menu selections, this time in depth. We learn how to create a rule, opening the Rulebuild window and entering rules in a simple, English-like method. The first box below the IF statement is the Object. The Attribute goes next to the right, and can best be defined as a characteristic of the object. The attribute in turn is best described by the Value.

In between Attribute and Value is the Operator. It characterizes the relationship between the contents of these two elements. Operators describe various relationships including:

verbal -
is, is not, are, are not.
mathematical -
less than (<), greater than (>), <=, >=, =,!=.
special (in THEN Result clause) -
Execute ($), Print (!), Show (*), and Parse(#).
These operators execute an AmigaDOS command, start a printing job, show an IFF image, and parse an equation to solve a dependent variable.

Also explained are threshhold and certainty values entered to the right of the values in the Premise and Result clauses respectively. A threshhold value of .20 represents the minimum degree of certainty a condition must possess before the related rule will be recognized by the inference process. A THEN clause that [dog][type][is][watchdog][.50] implies that there is a fifty percent chance (a 50% chance of being 100% certain) that the dog type is watchdog, IF it is determined that the [fur][color][is][brown][.20]. Clicking on the Auto gadget sets default threshholds at .20 and certainties at 1.00

which is good enough for most rules. Real-world problems rarely have absolute answers. Confidence factors, like threshhold and certainty, allow you to include rules that imply or suggest a solution, but are not absolutely definitive.

More on Cells

That reminds me: I promised to get back to cells, didn't I. By Selecting in the Cell option, you access the Cell Edit window. This contains boxes for the now familiar Object, Property (don't worry, it's a synonym for Attribute), Operator, and Value. Right and left arrow gadgets below the Object box are clicked on to locate particular OAV cells from the rule base currently in memory. Also displayed are gadgets labeled Legals, Variable, Inquiry, and Info. These buttons allow you to attach gadgets, files, and variables holding floating point numeric values to cell values. Selecting the Cell Display menu option helps you to see if values need to be cleared out before discrete inference processes, and, as with all edit windows, this one can be resized and put away by clicking on the box in the upper left corner of the display window. The program remembers all resizing and screen relocations.

Applying AI

Level Three, Getting Started, begins with some general suggestions for how to go about applying artificial intelligence techniques to real-world problems. It recommends using expert systems where the technology is useful and appropriate, anticipating the usage by providing users with easy, simple to use "handles", augmented with graphics, audio, and text explanation. It is strongly suggested to focus the problem tightly, breaking it down into smaller, specific problems. Many problems can then be handled with just a few rules. MAGELLAN can handle up to 100 rules in 512K, and will hold several thousand rules on a fully configured Amiga.

The tutorial is based on a diagnostic system of symptoms and possible failures of chips in the Amiga 1000. IF symptom is a black screen with a white bar OR symptom is a black screen only, THEN possible failure is Agnus Chip (8361). Rule entry is demonstrated with various screen shots and explicit directions. Be sure to pay attention to carriage returns, and make sure to click in the appropriate window before entering characters. (It is somewhat annoying to navigate between Dialog and Control windows via left button mouse clicks, but it is rumored that updates will advance the user interface to reflect the sophistication of the rest of the program.)

The various symptoms and appropriate failures are reduced from seven chart entries to four rules. The program automatically checks each rule when entered for recursion, so that backchaining does not send the program into an endless loop. Now it's time to test our first rule - first Clear All from the Session menu. This clears all certainties to zero prior to execution, and can be checked by Displaying in the Cell Option menu. If cleared, all cell values will be "unknown." Then we select the backchain option of the Inference menu. The Dialog screen will read "Backchaining Goal Entry: hit <CR> to begin." Click in the lower window and do just that; the system will now request the Object to reason about. It is looking for the Object of the THEN clause of our rule about [Possible][Failure], so enter "Possible" and <CR>. The system "knows" about possible failure, so it says "The system can reason to determine: What is the Possible Failure" and then "What Attribute do you wish to find out about 'Possible'?" We enter "Failure" and <CR> and a new window opens.

The Data Entry window is MAGELLAN's way of asking questions. It asks "Is the Symptom A Black Screen Only?" and if you click on the no button at the bottom, comes up again to ask the second IF clause of the original rule, "Is the Symptom A Black Screen With A White Bar Only?" If you click yes, you have given the system a Value that it can use to resolve the search for the goal, and the final result is given in the form of a Conclusion that is displayed in its own window: the Possible Failure is Agnus Chip (8361) [certainty = 1.000000].

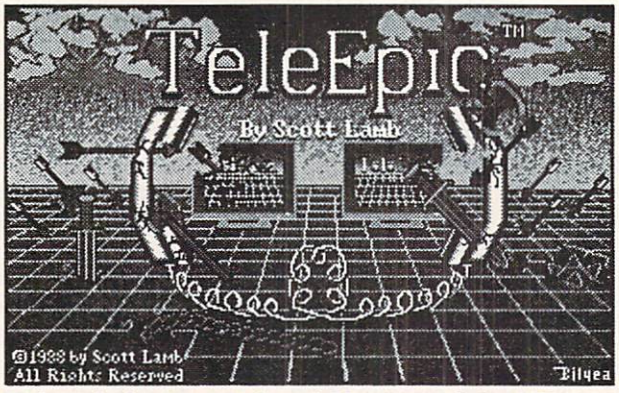
A Great Start

The tutorial concludes by showing how to bring up an IFF image in conjunction with the conclusion, and how to use MAGELLAN to generate legal values as suggested responses to questions. Included on the data disk are several sample knowledge bases that are worth loading and examining with the various Display options to see how rules

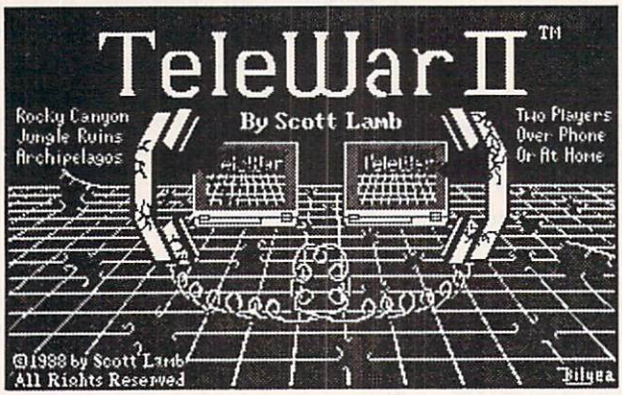
(continued on page 37)

THE ULTIMATE STRATEGY BATTLE SIMULATIONS

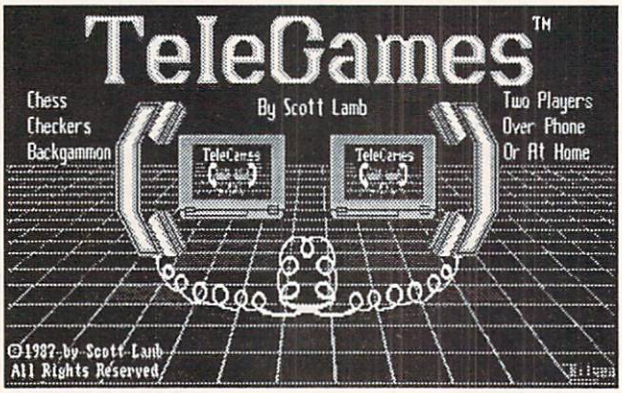
TeleWar, the *Ultimate Strategy War Game*, allows you to play against the *Ultimate Opponent*. Playing against an unpredictable human opponent assures you that victory is only sweet until the next game and that each TeleWar match is an exciting clash of intellect and intuition. **$39.95**



TeleEpic is the *Medieval Version of the Ultimate Srategy War Game*. Instead of commanding tanks and artillery, you lead a band of heroes waging war against your mortal foe. Test your skills wielding a mighty axe or sword. Throw a dagger or check your accuracy as an archer. **$39.95**



TeleWar II, the *Ultimate Strategy War Game Revisited* with new terrains and enhanced game play. Experience the ecstasy of blasting your best friend to rubble or disintegrate someone you don't even know. Watch out! You may be the next casualty peering out of a twisted pile of smoldering steel. **$39.95**



TeleGames, like all our '*Tele*' products, provides the enjoyment of competition without the problem of where to play. Competition is enhanced with smooth 3D perspective graphic animation and digitized sound effects. Players may compete on one computer or 'TeleConnect' via modem. **$34.95**

INTRUDER ALERT

The ***Intruder Alert Monitor*** allows your computer to become a monitoring, logging, or security system for your home or business. The system includes a startup kit including several types of detector switches and wire. Each of eight individual ports may be programed independently. Intruder Alert software allows for digitized audio alarms, speech synthesis, modem interaction, date and time stamping of events, external program launching, and utility support for BSR/X-10 controlling. Like all Software Terminal products direct support from the author and designer is available though our BBS service or via BIX vendor support area 'TeleGames'. **$159.00**

AVAILABLE FROM YOUR LOCAL DEALER!

SOFTWARE TERMINAL

3014 Alta Mere Drive
Fort Worth, Tx 76116
Phone: (817)-244-4150
BBS: (817)-244-4151

are constructed and how cells fill up as you Attempt back and forward chaining. You may have some difficulty in getting IFF pictures to come up, but make sure to direct the pathway to the data disk as KBases:Agnus, for example. Or load your images into RAM for faster display.

That's all folks!

That's as far as MAGELLAN goes in its Intro package. Once you've spent some time playing with the rules you've entered in the tutorial, and dissected the data disk examples, you may wonder what to do next. Don't discourage yourself by attempting too complex a system. Start by training yourself in how to reduce an everyday situation into a set of rules.

A good way of generating rules is by describing how you go about solving a problem. For example, let's say you want to figure out what to do tonight. There are a variety of options: Go to a movie, take a walk, get some Chinese food, stay home and use MAGELLAN to figure out this problem - no, that might be recursive! Each of these alternatives can have associated facts and rules of thumb. IF you like movies AND IF it's not raining THEN you might Go To The Movies .70 certainty. A further rule might relate to your preferences for comedy over drama, etc. You can see how quickly such a simple question can begin to generate rules. And then you have to factor in what your better half wants to do. You'll soon be developing an expert system on just who [wears the pants] in this house, let me tell you.

MAGELLAN, don't forget, is an Introductory package. It is also the first release of the first micro-based multi-tasking AI program. As such, it is strong on concept, deep in access and hooks to emerging technologies, but also tentative in interface and right-now application. This is as it should be; the Amiga user base is growing rapidly and many are new to computing in general. There is plenty to sink your teeth into now with MAGELLAN, and by the time you've learned how to choose an area for knowledge engineering and apply MAGELLAN to its domain, Revision 1.1 will be on the scene. Free to registered MAGELLAN owners for one year, additional features and improvements are promised including:

Arexx front end —

Currently, MAGELLAN can command the execution of Arexx macros via the '$' operator in the result clauses of rules. With the Arexx interface, it will be possible to use MAGELLAN as a "Knowledge Servor" to provide AI inference processing on demand from other, Arexx-supportive programs.

Inference loop mode —

continually 'waits' for new data and responds.

Real time clock —

automatic creation of 'Current hour', etc., cells and the upkeep of their current values. In combination with loop mode, a powerful feature.

Improved Rule-Management Tools —

better display, editing features

More 'Meta-Logic' —

greater control of MAGELLAN features from within a knowledge base itself — to allow users to control knowledge base loading, clearing values, etc., under rule control.

These revisions will go a long way toward answering the questions that need to be asked of MAGELLAN. The appearance of AI applications in the Amiga market is significant; there is an enormous body of research and commercial application in expert systems available for study and incorporation.

Summary

With the proliferation of "main-frames on a chip" goes hand-in-hand an explosion of information and the need to manage it. Expert systems are here to stay; they're embedded in your VCR, your coffee-maker, your cash machine. MAGELLAN gives you the keys to the future, as in: Gentlemen and women, start your [inference] engines!

MAGELLAN
Emerald Intelligence
334 South State Street
Ann Arbor, Michigan 48104
(313) 663-8757
$195.00

•AC•

E.C.T. SampleWare

Incredible multisampled sounds in SoundScape and IFF formats

by Tim Mohansingh

Long before MIDI, and even before the personal computer, there was the first sampling keyboard. It didn't have a single byte of memory yet it could play back samples seconds long. It was the Mellotron, played avidly by many, including the Beatles and the Rolling Stones. Just listen to Days of Future Passed (The Moody Blues) or Strawberry Fields Forever (The Beatles) to hear the "mutative organ". People reproduced anything by recording a sound onto a dozen closed loop tapes (it was the first multisampler, too!) and controlling the tapes with a Rhoads sized keyboard.

The technique to play notes ranging between the twelve samples was the same as it is now. The tapes were played faster to get higher notes. Today, tapes and speeds are replaced by bytes and sampling rates. A recent introduction to this modern musical software corps is E.C.T. SampleWare, created by Drew Neumann and Todor Fay, the programmer of SoundScape. The samples from E.C.T. were generated with the Mimetics SoundScape Sampler using the SoundScape Pro Midi Studio. Four disks are currently available in either SoundScape or IFF format: Rock, Orchestral, Grab Bag and Digital Synthesis, containing a total of 88 samples. These will be reviewed here along with some comments on their application and background.

Why sampled sound?

Sampled sound has a distinct advantage over traditional synthesis because the character of the sound doesn't depend on a particular kind of hardware or software. Not only can you reproduce traditional stationary tones, but complex, evolving patterns can also be created. When the sound has a regular pattern, it can be looped to create a continuous sound, a feat difficult to accomplish with traditional synthesizers. The E.C.T. samples contain several gems that take advantage of this ability. ElectroBoil and ElectroBubble are two examples that sound exactly as they read.

Measuring up

The piano imitation in any sample package (or synthesizer) often serves as a good measure of the instrument's overall quality. The piano for E.C.T. was multisampled for 5 octaves, and like all the other samples, has good fidelity. The E.C.T. piano is over 50% bigger than the IFF piano sample from Deluxe Music, but shows considerable improvement in character (thanks to multisampling) and less aliasing distortion (the benefit of a higher sampling rate). The envelope settings needed a little adjustment, however, when first loaded, the velocity sensitivity was zero (not good for a piano with any feeling) and the attack a little too sharp. Softening the attack was necessary to eliminate some bothersome clicking.

MegaSounds

I usually don't get too excited when somebody hands me a grab bag of sounds because it's difficult to find a practical application for a nutty variety pack. In Grab Bag, though, there are many noises that actually sound musical. Careful thought went into the production of masterpieces like Glug Glug, Wet Blorch and Wacky Flower Pot. Grab Bag has sounds that will work as the spice for your music, or even as alternative rhythms.

Digital Synthesis is the newest addition to the E.C.T. SampleWare collection and has some of the most gutturally impressive sounds. The deep vibrations coming from Gnarly Stack and Phlangarama make the Amiga sound much bigger than it looks.

Rock has pretty much what you would expect, samples of all sorts of guitars: stacked, fuzzed, clean, acoustic and synthetic. The drum kits are excellent. Orchestral is also pretty straightforward with wind instruments, strings, horns and percussion. Orchestral Hits is a very special sample that is guaranteed to keep anyone awake during your music. It's a snapshot collection of the "bangs" where an entire symphony is blasting the same staccato note. Pretty impressive. Overall, Orchestral demonstrates well the night and day difference between professional samples and home samples.

What do you get?

On the outside, the E.C.T. package comes with a 7 page booklet, and the four descriptively labelled disks. The booklet describes what went into making the samples, how to use the samples with your system (whether it be SoundScape or an IFF compatible application), and how to custom tailor the samples to fit in less memory. It also describes how to use SoundScape to create a vibrato effect.

All this and IFF too!

For folks who don't have SoundScape and want to use the IFF samples with other software, E.C.T. also offers multisampled sounds in IFF format! Using proprietary software written by Todor Fay, separate sounds were recorded on different octaves and combined into one sample. Multisampling preserves the quality at higher and lower octaves for wide range instruments like the piano, and is indispensible for creating a good drum kit. The IFF samples also have icons allowing them to be dragged into different directories without using the CLI. This is handy for people with Instant Music and other software that need sounds in a special instrument directory. Sonix users will need to rename their samples with the suffix ".instr". In addition, Sonix users will need to play with the keyboard note assignments in order to hear the sounds once they are loaded.

The Making of a Sample

Creating a good sample takes a few tricks: First, good audio equipment, and then careful attention to sampling rate, envelope and loop point. Unfortunately, even with the most careful setting of the looping point, nasty clicks can show up in the sound. Todor Fay got rid of this problem by using a crossfade module in SoundScape (see Amazing Computing volume 2, number 9). By crossfading the beginning of a sample with the end, smooth transitions occur as the sound is held. Most of the time crossfading works well. It was used heavily with the E.C.T. sampleware to ensure that the samples could be sustained without distracting attention.

Make those sounds more interesting

All sampled sounds have an inherent disadvantage, they sound exactly the same every time they are played. That's why it's hard to sample a saxophone and fool anyone into thinking what you then play is live. Quite often, acoustic instruments depend heavily on drastic variations to make the sound alive and to convey a human feel. So when a guitar or flute is sampled and used to play a sequence, it can be pretty dull. Fortunately, there are several tricks you can use to make samples sound more interesting.

1. You can add a vibrato to a sound in SoundScape by using a pitchbend track and the Tapedeck. This is explained in the manual that comes with the samples, and an example track sequence is provided on the SoundScape version sample disks. Vibrato generated this way is very realistic since it is independent of the sampling rate. That means no chipmunks as you play different notes, the vibrato rate doesn't change. NICE.

2. The Echo module in SoundScape (see Amazing Computing™, Volume 2, Number 5) does wonders for a sound by improving the depth and hiding little glitches in the sample. Although it's expensive in terms of the audio resource (you may not be able to play more than one sample when using Echo), this effect can turn an otherwise dull instrument into a great solo. Synth 4 on the Rock disk sounds incredible with an Echo effect.

3. If you don't want to go as far as the Echo module, things can still be improved for lush sounds by lengthening the envelope to get soft fades. Here again is a compromise between smooth sound transitions and running out of audio channels.

4. Pay attention to your playing style. Hold notes just long enough to hear the character, then let go or play something else before looping becomes obvious. This may be hard to do if you are playing a piece by strict time, but keep it in mind when composing.

You may find that just as everything is fattened up to where you like it and you're ready to add the drum track, you've run out of audio channels. The next step is to add an external synthesizer and control it with MIDI to get the extra sounds. Alternatively, the combination of an inexpensive synthesizer and the Amiga playing notes together (called stacking) can make very impressive single sounds that imitate much more expensive keyboards.

Sampling rates and filters

For someone with lots of memory and a picky ear, sampling rate becomes an important issue. Sampled sounds on the Amiga have until now depended on a hardware filter to eliminate aliasing distortion caused by a low sampling rate. Now the filter can be switched out with a software button and many want to hear the higher frequencies that the Amiga is capable of reproducing. The E.C.T. samples were sampled at 14 kHz, which is a reasonable compromise between high fidelity and nicely sized samples. Another solution which would allow the user to make compromise decisions would be to sample the sounds at the maximum rate (about 28 kHz) and then provide a filtering program that would allow sample sizes to be reduced when memory is short.

Good stuff!

The bottom line criteria for a good sample is musical utility. Answer the question: Can I really use this sample? E.C.T. SampleWare was designed with this principle in mind. The sounds were professionally recorded and well groomed characteristics that are hard to get with a home sampler and little patience. If you're interested in trying these samples for yourself, send an order with $24.95 for one disk or $79.95 for all four to:

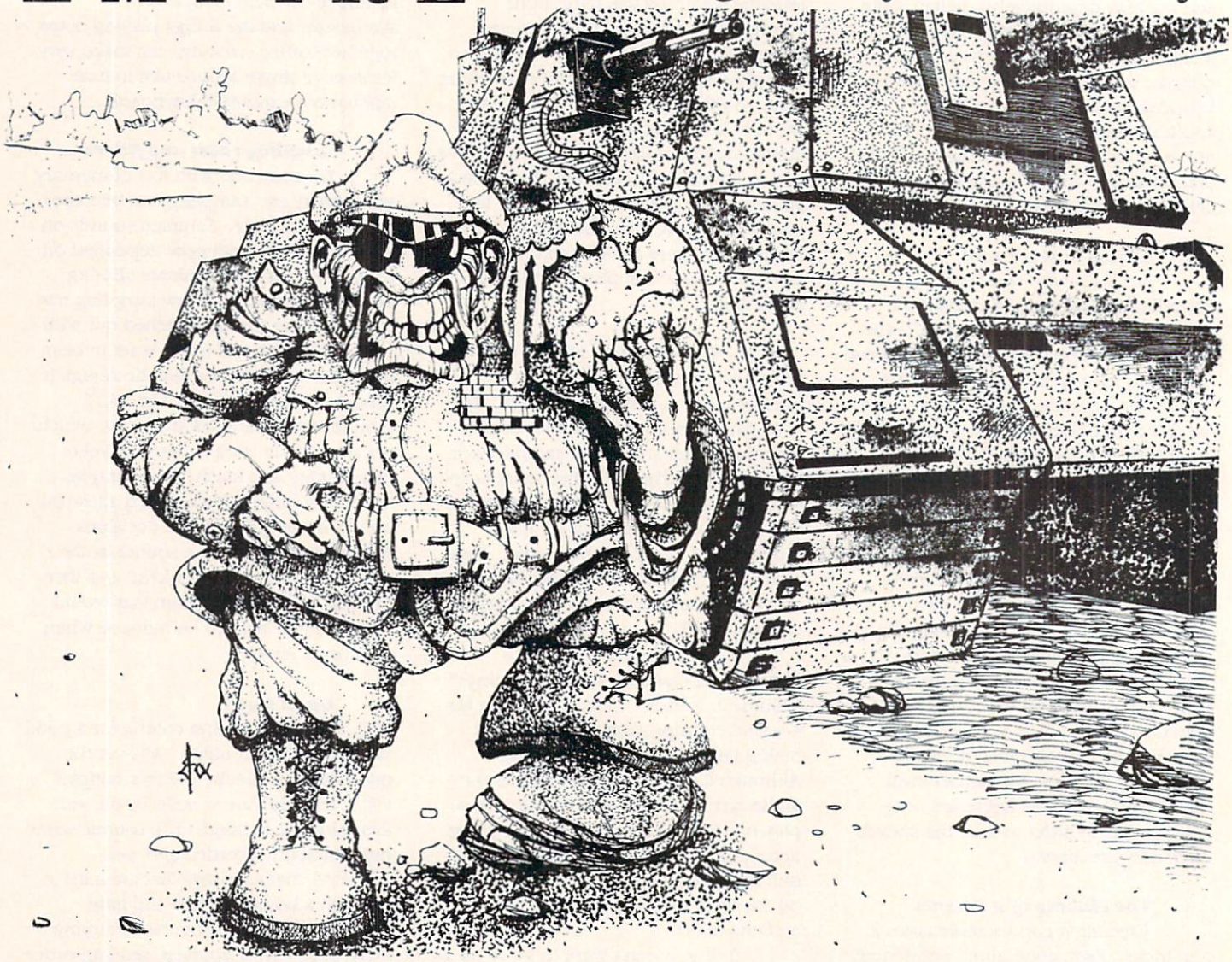
E.C.T. SampleWare
PO Box 36
Sierra Madre, CA 91024

Don't forget to specify whether you want the IFF or Mimetics version.

•AC•

EMPIRE

Wargame of the Century



by Stephen Kemp, PLINK: SKEMP

EMPIRE has finally made it to the Amiga — perhaps one of, if not, THE greatest computer game ever written! I have waited and waited for this game to appear on the Amiga. Now that it has, I am not disappointed!

Ancient History

Many of you may be familiar with previous versions of EMPIRE that were available on other machines. EMPIRE was originally written by Walter Bright while he attended Caltech. Later he rewrote the program to run on the PDP-11. After that, many people became "addicted" to the game and it was destined to migrate to other machines. A version of EMPIRE was translated into C for the IBM PC around 1984. Although that program was being marketed, like all popular games, imitations sprang up in the public domain. Mr. Bright and the marketers of EMPIRE may have been "annoyed" by these unauthorized duplicates, but it may all turn out to their advantage. Once it becomes known that this version of EMPIRE is the greatest yet available, EMPIRE "addicts" will not be able to resist.

The Warning

Yes, EMPIRE is "addictive". I have been an EMPIRE addict for many year. When I heard that INTERSTEL's EMPIRE was going to be distributed by EA, I thought how appropriate....EA — Electronic Arts or maybe EMPIRE Anonymous. Apparently, INTERSTEL was concerned enough about the addictive nature of the game to include this warning on the box:

"WARNING: This program is highly addictive. Considerable otherwise productive time might be lost. Play only during vacations."

Do not take this warning lightly. Believe me, once you start this game, all sense of time is lost. Hours will pass by in seconds and days will pass by unnoticed. Before you know it, you will be just like me, an EMPIRE addict. You will look at your watch at 3:00 AM and say, "Oh, I'll go to bed just at soon as I do this one thing." And after that one thing is accomplished, you will say it again, and again.

About EMPIRE

EMPIRE is a game of conquest. INTERSTEL has done a nice job of working the scenario into its Star Fleet saga. You are sent to a planet to conquer the inhabitants and increase the reign of the EMPIRE. However, one or more of your EMPIRE's enemies have the same intentions. Now it is more than a simple conquest, you must also eradicate your enemies.

Planetary conquest is accomplished by bringing the planet's cities under your control. Once you have cities under your control, they can be directed to build armies, fighters, and ships so that you can expand your control and plan to meet your enemy. When you finally do meet the enemy it will be a test of strength and endurance. Ultimately, the one who makes the most strategic use of their resources will be the victor.

EMPIRE is not disk copy-protected but it is play-protected, which means that you are annoyed at the beginning of each session by having to type in a word from the documentation. The only problem I have with this protection method is that nobody seems to implement it properly. When they ask you for a word, they give you the word number, the line number and then the page number. When you look it up, you need the page number, the line number and then the word number. Am I the only one who has noticed that they give it to you exactly opposite?

War Pieces

In order to take over a planet and defeat your enemies you will need a number of war units. Here is a brief description of the pieces that you can make during a game:

Armies — These are the most important units when conquering a planet and fighting a war. Armies can be produced in the least amount of time and are the only piece that can conquer cities. This makes them indispensable, since you need cities to make more weapons.

Fighters — Unsurpassed in their ability to quickly discover new territory, the fighters are also good in combat. This is due to the fact that fighters can "move" a greater distance in a single turn than any other piece. This gives them the ability to get to the front quickly once the enemy is detected.

Troop Transports — Troop transports are used to move armies between continents. They are not well suited for combat, but will perform when necessary.

Submarines — One of the most useful battle units in the sea is the submarine. They are hard to detect by your enemies and can inflict tremendous damage when they attack other vessels.

Destroyers — These are the fastest ships. Because of their speed, they can be sent to enemy sightings quickly. Destroyers are good in battle and as escorts for transports.

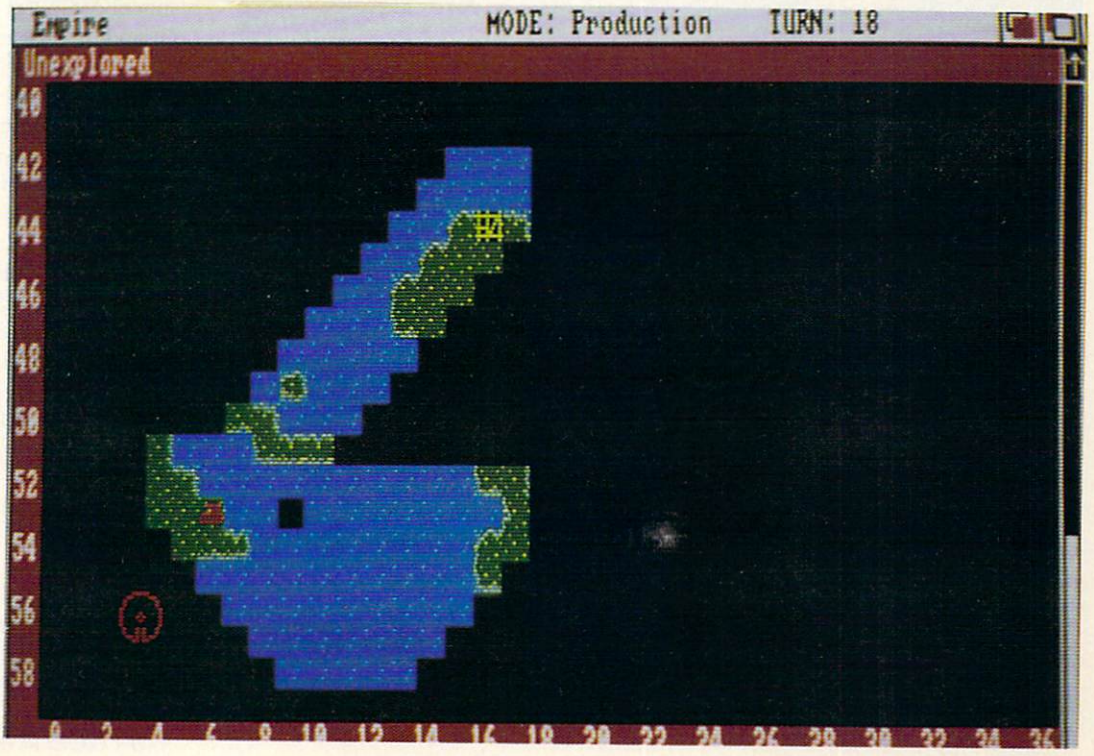
Cruisers — Cruisers are not as fast as destroyers but are much stronger. When engaging the enemy at sea, the cruiser is a good weapon.

Aircraft Carriers — These are large sea vessels which can carry up to 8 fighters. Since fighters require periodic refueling, it is sometimes inconvenient to send them back to a city. Carriers can be used to refuel fighters and, unlike cities, they can move to where the action is.

Battleships — The final and largest unit available to you is the battleship. They are heavily armored and almost indestructible when engaging a single opponent (unless it is another battleship).

Of course each unit has some disadvantages. The pieces might be thought of in the same manner as the old "Paper, Rock, Scissors" game. Some pieces are not as effective against one type of piece, while extremely effective against others. Additionally, you have to wait for a city to build the units. I have listed them in the order of the amount of production time required to make each item. Armies require 5 turns and battleships require 50. A lot can happen in the 50 turns it takes a city to produce a battleship. You may have needed the 10 armies that could have been produced in those 50 turns. Also, by the time you discover the enemy, the front could be so far away from the city making a battleship, that it might not get there in time to help. Don't let the production time prevent you from making strong pieces, just keep in mind that when you meet the enemy, you want to have as large a force as possible. That means you will want to have plenty of armies, fighters and cities. They are the foundation of a strong front.

(continued on page 46)



Building a map of the world, in the Production mode.

The Making of Don Bluth's Dragon's Lair

by Randy Linden

Insert Coin

In the summer of 1983, a revolutionary new game appeared in the arcades. This game featured high quality stereophonic sound, the likes of which had never been heard in the arcades. But even more importantly, this game dazzled the arcade going public with it's high quality cartoon style animation. This game was, of course, Dragon's Lair, and the arcade scene has never been the same since. Dragon's Lair ushered in a new age of high quality graphics for both laser disk and nonlaser disk games alike. Gone are the rudimentary graphics of early arcade games to be replaced by dazzling, realistic displays.

Seeing Dragon's Lair in an arcade first seminated the notion of high quality graphics on computers. It seemed only natural that similar results could be achievable on a personal computer. With that thought in mind, the hunt was on for a computer which could keep up with the demands of real time full screen animation. As it turned out, however, the search was difficult and the wait lengthy.

Dragon's Lair on a C64?

Visionary Design Technologies' first real attempt at recreating Dragon's Lair on a personal computer was undertaken using the Commodore 64, then, the most graphically sophisticated personal computer around. After lowering expectations time and time again to allow for hardware limitations, it was concluded that satisfactory results could not be achieved and the idea was put aside to collect dust until a revolutionary new computer came along.

This revolutionary new computer did come along in 1985 in the form of the Amiga which astounded everyone with it's graphic and audio capabilities. However, it was not until two years later, when the Amiga had matured sufficiently, that the project became truly feasible. While the actual computer was there, there was neither the support software nor sufficiently powerful hardware to aid in the development of the game.

Why wait for the Amiga?

The Amiga's most important trait, from the point of view of the game, is the incredible color graphics. While many other computers are capable of achieving

(Top and Bottom) Scenes from the Amiga version of Dragon's Lair.

similar results, few can do it as well, and even fewer can claim that every owner will be able to achieve such results. Further, the Amiga lends itself to graphically intensive operations because of it's blitter coprocessor. The blitter can, in effect, take over some of the more mundane operations from the microprocessor and, as well as executing them faster than the microprocessor, frees it up to work on more sophisticated operations.

It is this splitting of responsibilities between the microprocessor and the blitter which allows for the very advanced compression techniques that are used throughout the game. Similarly, more efficient and better sounding audio compression is made possible by this unique combination. Unlike the industry standard IFF format with it's limited compression abilities, the VDT compression formats push the machine to it's limits. These new formats achieve significantly better results with a combined increase in play speed and a decrease in storage requirements.

Animation with CAST

The Amiga's unique blend of inherent graphic abilities and processing speed makes it ideally suited for animation. It was this combination which convinced Visionary Design Technologies to begin working on a powerful Cel Animation Sequencer Tool, code named CAST. This program would allow cartoonists to easily animate with their computer while emulating some of the processes they were accustomed to from classical animation techniques. CAST may yet be released commercially if sufficient interest exists within the Amiga community to justify the completion of this very large project.

It was while trying to come up with a creative demonstration of the power of the CAST package that the idea of resurrecting the computer version of Dragon's Lair, now under the official title of Don Bluth's Dragon's Lair, came up. From our point of view, this course of action had two advantages. First, the animation was already done, so we had only to digitize and touch up the actual animation frames. Furthermore, the product would have an easily recognizable animation sequence to show off it's power. Don Bluth's animation is recognized internationally as being of the highest quality, and is a perfect compliment to the technical excellence of the CAST animation system.

Looking back

Our first step was to research the market for all currently applicable hardware and software to make the task at hand easier. After an exhaustive examination of all image and sound capturing hardware available at the time, we finally settled for Sunrize Industries' Perfect Vision and Perfect Sound digitizers. Of the many digitizers we looked at, their hardware simply outperformed the others. They were faster, easier to use, and the results were much better. The people at Sunrize Industries were very helpful with their excellent support, and they produced custom versions of their digitizing software suited to our needs.

Digitizing and animation

Once we had picked our hardware, we could get on with the actual job of producing Don Bluth's Dragon's Lair. The processing of the animation is actually quite tedious as it involves digitizing each frame from the laser disc, and passing it through the hands of an Amiga artist who "lifts" the animated parts off the background. That is to say, the artist replicates a picture which contains only those images that are currently moving in the animation. This is necessary because, even with the fine digitizing equipment we had available to us, no two frames come out exactly the same. This lack of consistency would have caused all the animated characters to appear as though they were shimmering. Other artists are assigned the task of reconstructing the background for the given animation sequence. This process is very artistic in nature, and requires a good knowledge of both classical art techniques as well as the limitations of the Amiga. It is up to the background artist to create the illusion of depth and perspective for the animation, because the animated characters are very two dimensional. Once both the foreground and background pictures have been cleaned up, they are combined and sequenced.

The work now passed into the laps of the four rotoscopers who put both sets of imagery together, and made sure it was consistent. Along with combining the foreground and background art, the rotoscopers were responsible for making sure all the trivial rules were observed; Such things as matched color palettes for a given sequence, centering and the like, are all important for the final product to look both clean and professional.

Now for a little music

Once the animation sequence is established, it has to be synchronized to the audio track. While this process is fairly straightforward for rooms which were transferred directly from the original game, rooms which needed to be edited required some audio manipulation. As each animation sequence was adjusted, the sound had to be resynced with the video tracks so that everything remained consistent. When all the sound sequences were finalized, we went into the studio and digitized the complete sound track for the game with the perfect sound digitizers. The audio was compressed, and combined with the animation to form the finished rooms.

Creating the code

While the artistic process was going on, there were also programmers working out the technical wizardry which needed to take place before Don Bluth's Dragon's Lair on the Amiga could become a reality; making the Amiga run faster and more efficiently than ever before. As well as creating the software which generates the game, the actual game, and all of the compression algorithms needed to be designed and implemented, it was decided early on in the project that we would take the original Dragon's Lair concept and extend it beyond it's arcade rendition. The result was the incorporation of an entire arcade game in the computer version. This maze section pits a smaller animated version of the player against monsters in a large, omnidirectional scrolling maze. The player must find his way through the maze and pass through each of the animated rooms that are behind the doors in the corridors. The maze portion of the game and all of it's support software needed to be written

(continued)

complete the work. A total of four programmers took part in the effort with the overwhelming majority of the programming time spent on the generation software and player portions of the game.

Another programming first!

This game is also unique in that it accomplishes several programming "firsts" in the Amiga community. Aside from the fact that this is the first ever cel animated game for any computer anywhere, it is also the first ever fully over scanned video game for the Amiga. Add to this it's high resolution and interlaced modes, as well as stereophonic sound, and you have a dazzling product. There are also several firsts which are transparent to the user. This is the first piece of software which can utilize the Writable Control Store, or in plain English, the 256K of RAM found in all Amiga 1000's but not normally accessible for anything but Kickstart. This, effectively, gives Amiga 1000 owners with a 256K expander 768K of RAM. That is why the game requires an Amiga 500 or 2000 with 1 Meg of RAM, but only a 512K Amiga 1000. Another transparent but valuable feature is concurrent loading; while the game is being played, it is also transparently loading information from the disk. This is accomplished without any loss of speed in either animation or audio, and allows smooth running animation much larger than available memory to be played with no interruptions in game play.

Because of our commitment to support as many Amigas as possible, we had very serious size limitations on the player code which actually executed the program. It is, of course, written completely in 68000 machine language, but was written and rewritten many times to improve efficiency and speed, as well as to shrink it down. Similarly, the generation software is also written entirely in 68000 as is the maze portion of the game.

Finding the talent to make it possible

We realized quite early on that we would quickly exhaust normal channels for finding artists when our requirements were so high. We decided to be true to our usual form, unconventional and find new talent through local BBSs. We decided to opt for local BBSs for obvious organizational reasons, but we would be happy to hear from any of you who are interested in working on future projects. In our experience, many people who may not have considered working as artists (or programmers, in fact) actually make for a great addition to the development team if given a chance. The response from the BBS public was tremendous, and it is largely due to the help of the various users and SYSOPs that this project was completed.

Of Drawbridge rooms and Hard drives

When the game was first introduced to the Amiga market place at a preview at AmiEXPO Midwest '88 in Chicago, the two most frequently asked questions were "Is the room no one ever saw in there?" and "Will there be hard drive support?". The first question refers to the Drawbridge room which was not seen on most of the Dragon's Lair arcade machines, and the answer to that question is yes. The second

answer was, unfortunately no, or rather, a qualified yes. With the obvious need for hard drive support we added this problem to our list of programming challenges to be overcome. We soon realized that to include any type of hard drive support we would require large amounts of time and cooperation from the hard drive manufacturers. A thorough understanding of each hard drive's software and hardware is necessary before it can be incorporated into the program. Actual hard drive code must be inserted into the game and this requires that hard drive manufacturers release to us their source code.

With the help of the fine people at Comspec, and their great hardware, we set about making it possible. Not only is the game installable on their hard drives, but this installation is transparent to the user. The user has only to insert the master disk, and the game will automatically adjust to any Comspec device that happens to be present. This means that not only are hard disk drives supported, but also Bernoulli boxes as well as any other SCSI device. We are hoping to add other hard drives to our list of supported devices, but cannot do so without the help of the individual manufacturers. At this time, only the Comspec SCSI controller is supported.

Game Over

When compared with other games available for the Amiga, and there are many fine entertainment products out there, Don Bluth's Dragon's Lair is indeed a ground breaking product which may irreversibly affect the standards by which Amiga games are judged. With it's real time animation and sound, combined with it's exciting arcade gaming, Don Bluth's Dragon's Lair promises to be a big favorite for the foreseeable future. As Michael T. Cabral wrote in his AmiEXPO Midwest '88 report, "When you first see the graphics and animation, you'll take a look around to make sure you didn't somehow stumble into a video arcade."

Versions of Don Bluth's Dragon's Lair are currently under development for other popular personal computers, but very few of those machines appear to be capable of handling the enormous amount of work the Amiga must accomplish in order to achieve these spectacular results. Whether or not other versions are released will depend largely on the success of our programmers in pushing machines which were not designed for the task of animation to that lofty goal. With the Amiga setting the standard by which to measure our success on other machines, we can make no promises right now.

On behalf of the whole Don Bluth's Dragon's Lair development team, we hope you enjoy this game.

We would love to hear from any programmers or artists out there who would like to work on Amiga projects, or from those of you who have projects in mind you would like to see.

For more information contact:

Visionary Design Technologies
45 Whitehorn Cres.
Willowdale, Ontario, Canada
(416) 492-9954

•AC•

*(**EMPIRE**, continued from page 41)*

Playing EMPIRE

If you have ever played EMPIRE, you will have no trouble playing this version of the game. For the novice, never fear, the documentation is good and the menus are easy to use.

There can be up to three forces vying for control of the planet. Any or all of the forces can be controlled by players or the computer. EMPIRE offer a play-by-mail option, but modem play is not available (yet). Each force has a different color, so they can be easily distinguished. The playing ability of the computer can be changed as can the combat and production efficiencies of the game pieces.

The world is usually divided up into several islands or continents. Your map of the world is divided into a 60 x 100 grid. To begin the game you are given a single city. Eventually you must produce an army from that city if you want to conquer another city. Each unit (including cities) only knows about its immediate surroundings. Any area of the map that you have not yet visited will remain black until you move into that area. As your units move around, the map will be filled in with the information relayed from them. However, once your piece moves away from an area, the only thing that is certain is whether the area was land or sea. The enemy can and will be moving, just as you are. Opponents can only be seen while they are next to one of your units, but you will be able to tell where you last sighted them.

To attack the enemy or a city, simply attempt to move your piece onto the occupied square. Then the battle begins. You never know who will be the victor of an engagement. It depends upon the fighting ability of your piece, the defensive ability of the opponent's, and a little luck.

You control your pieces by giving them "orders". Some pieces can be sent out on random, while any unit can be given a direct order as to move to an exact location or in a general direction. If a piece cannot follow its orders it will wait for you to help it out, but once you get the hang of giving orders this will seldom happen. Orders can be given to pieces via the keyboard, mouse, or menus.

Improvements

One of the most impressive improvements of this version of EMPIRE over all the others is the inclusion of sound. I could not believe how much difference it made in the game. The armies' tanks rumble, the fighters buzz, and the ships swish (?). At first I thought the sounds would get old, but now I can't play without them. If you feel otherwise you can turn the sounds off. When you take weapons into battle, you hear the boom of large guns while the engaged pieces flash on the screen. When the guns go silent there will be only one survivor.

The graphics are also an improvement over the previous versions of EMPIRE. As you might expect, land is green and water is blue. Unlike older versions of the game, the war units are no longer simple letters. An army looks like a tank when active and a row of tents when on sentry. Fighters naturally look like planes, and the ships all have different designs. It is supposed to be possible to tell whether a troop transport or carrier has troops on board, but I found it fairly difficult. Aside from this short coming, I found the graphics excellent.

Creating Worlds

Finally I should note that you can create your own worlds to conquer. There is a map editor included with the game. A number of pre-designed maps are also included, so you won't have to produce a world right away. Once you have played with these worlds long enough, you may be able to start recognizing them. Unfortunately, the program only randomly selects a game map, it will not randomly generate one.

Final Words

The point of EMPIRE is to be the sole survivor. It can only be accomplished by careful planning. This version of EMPIRE is easy to play and hard to put down. Once you get your hands on this game you won't care about anything else. In fact, it was almost impossible to write this review, because it tore me away from my game.

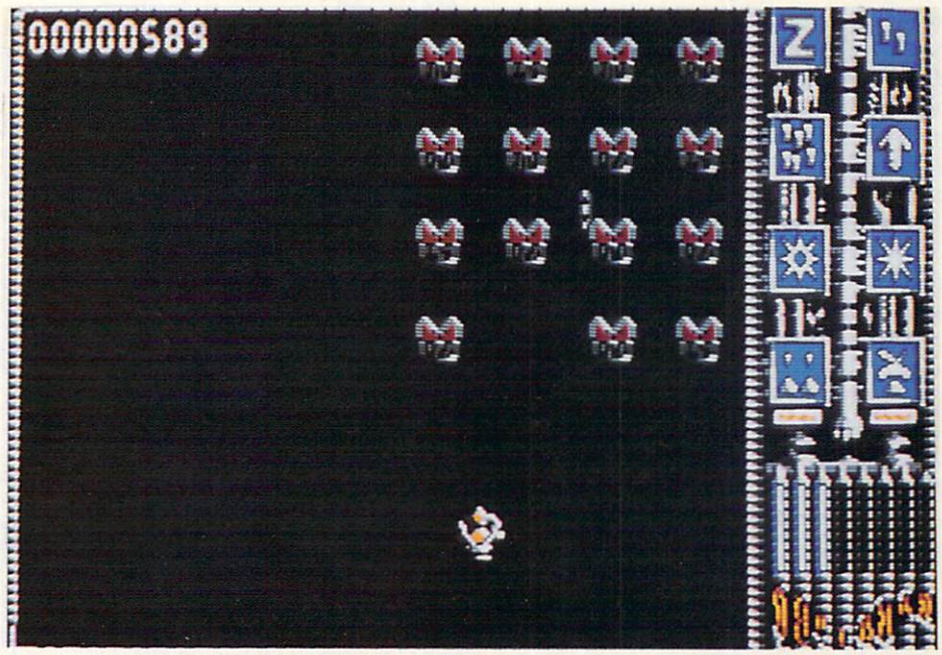
EMPIRE *$49.95*
Electronic Arts
1820 Gateway Dr.
San Mateo, CA 94404
(800) 245-8525; in CA (800) 562-112

•AC•

BETTER DEAD THAN ALIEN

by Jeffery Scott Hall

Better Dead Than Alien, from Electra, is being exclusively marketed in the U.S. by Discovery Software in their first attempt at affiliated labels. After the game has loaded, you will find yourself at the main options screen where you will find four options which are: Start The Slaughter, One/Two player mode, Input Options, and Level: Practice. If you wish to change your input device for game play, then you should select the Input Options from the main screen. This will allow you to select either mouse, joystick, or keyboard for both ports one and two. The last option on the screen, Level: Practice, will allow you to enter the name of the level you wish to practice. Since each level in the game has a name, you may also use this option to skip past the levels that you've already mastered. Once game play has started, you will find yourself surrounded by wave after wave of mean and ugly aliens.



Mom always said there would be days like this

You assume the role of Brad Zoom, a very happy space exploration guy until he landed on Mars. Suddenly, you find the ever so peaceful planet to be corrupted by aliens inside of what they refer to as the "battle zone", which is appropriately named. Your job is to enter the battle zone and destroy as many aliens as possible as they come at you laser ready. At your disposal is a rocket ship fully equipped with lasers. You have full control of the ship which can be moved left, right, up, and down the screen. The game will start with your ship at the bottom of the screen, and three tanks of energy (or lives) for shields. You will find your ship to be flying over a vertically-scrolling background which contain some of the best visual effects I have ever seen in a shoot'em-up. Aliens will enter the screen in many different attack formations, causing their defensive and offensive powers to be very strong. They will start at the top of the screen and slowly drop down to the bottom. As they are doing this, one of the aliens will go for what I call a "suicide dive" in which it will drop from the attack formation and go directly for your ship. Of course, while all this is happening, they are firing a barrage of lasers aimed directly at you.

You will find Better Dead Than Alien to contain a whopping 72 levels of play, each one with completely different aliens, attack formations, and backgrounds. As you progress from level to level, you will find each one more challenging than the last. I have been playing the game for about 30 to 35 hours, and I still have not made it past level 10. This is due to what they call the "bonus rounds". I call them the "freeway express rounds" because they require you to have the reflexes of a Sunday driver on a downtown expressway.

(continued)

Bonus rounds?

After you have managed to successfully destroy three waves of aliens, you will be presented with one of three different bonus rounds which are: Rocky Ride, Master Alien, and Super Alien. These have got to be the toughest bonus rounds I have ever seen for a game. They claim you can rack up some high scores here, I claim you can panic! The last two, Master Alien and Super Alien are just large aliens which are almost indestructible requiring you to have fast reflexes as you attempt to destroy them. However, the Rocky Ride is a different story all together. This round contains three very large boulder aliens with mean looking eyes that when hit split into smaller boulders. The best way to describe this is to think of the old Atari classic Asteroids, increase its speed by twenty times, and you have a good idea of what this bonus round is like. These are just a few of the bonus rounds you will encounter in the game, others await at higher levels to really challenge your reflexes.

Don't fire until you see the greens of their eyes

That's right, aliens don't have white eyes so you must fire when you see their green eyes. What does this mean? Well, it certainly doesn't mean you're flirting with an outer space creature, but it does mean you've caught it at the weakest moment. When this happens, a power capsule will float down towards your ship for you to pick up by flying over it. The panel icons are to the far right of the screen, and depending upon which one is lit up at the time you pick up the power capsule, an enhanced feature will be given to your ship. The laser scatter bolts allow your ship to fire multiple lasers, auto repeat saturation blasts will destroy multiple hit aliens with one shot, stun effect causes the aliens to freeze and stop movement, and clone ship doubles your ship (not lives) for added fire power. Others include multiple blast which fires a burst of missiles, armour missile takes out an entire rank of aliens, neutron bomb weakens all the aliens, and shield gives you limited protection from enemy fire. If one of the panel icons is not lit up at the time you pick up a power capsule, an extra energy tank will be given to you.

Summary

After playing the game for quite some time, I have found only one major complaint. That is, the inability to store the top ten scores to disk. If you get a really high score, you will be asked to enter your name, and when you stop playing the game you will lose your score forever. This hardly seems right, especially when you want to boast your score of over 500,000 to your friends. Hopefully Electra will change this in a future version of the program.

Better Dead Than Alien combines a great blend of arcade-action, superb graphics, and sound effects to make it the best shoot'em up I've seen. Due to the many levels of play and changing scenery, you will find it difficult to get bored with playing the game. If you like lots of arcade action where your fire button is used to squeeze off millions of shots, then this one is for you!

Hints and Tips

1. Never stay in one spot for more than several seconds. If you do, it will give the aliens an easy target for them to fire at and go for a suicide dive.

2. Make sure and collect any power capsules that are available, for this is the key to making it out of really tight situations. If your timing is right, you can wait for the panel icon lights to go off before you have to shoot the alien with the green eyes. By doing this you will be given an extra energy tank (or life). Be careful though, not reacting quick enough could result in no power capsule at all.

3. When you encounter the Rocky Ride, use the following techniques for survival. You should concentrate on only one boulder alien at a time. Shoot this one first, then keep shooting the smaller boulders until it is gone. Repeat this process until you've made it through the Rocky Ride.

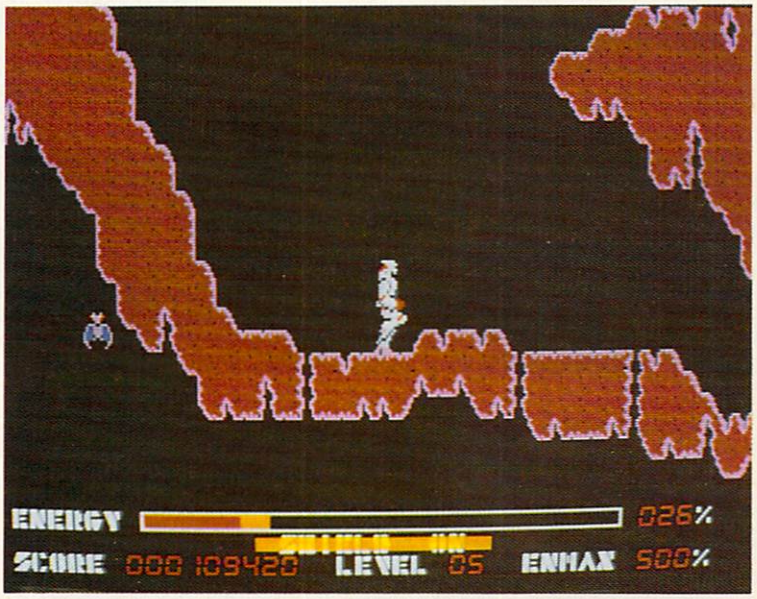
Better Dead Than Alien (Amiga)
Electra Software

Distributed in the U.S. by:
Discovery Software
163 Conduit Street
Annapolis, MD 21401
Phone: (301)-268-9877
Price: 34.95

•AC•

THEXDER

by Bruce Jordan



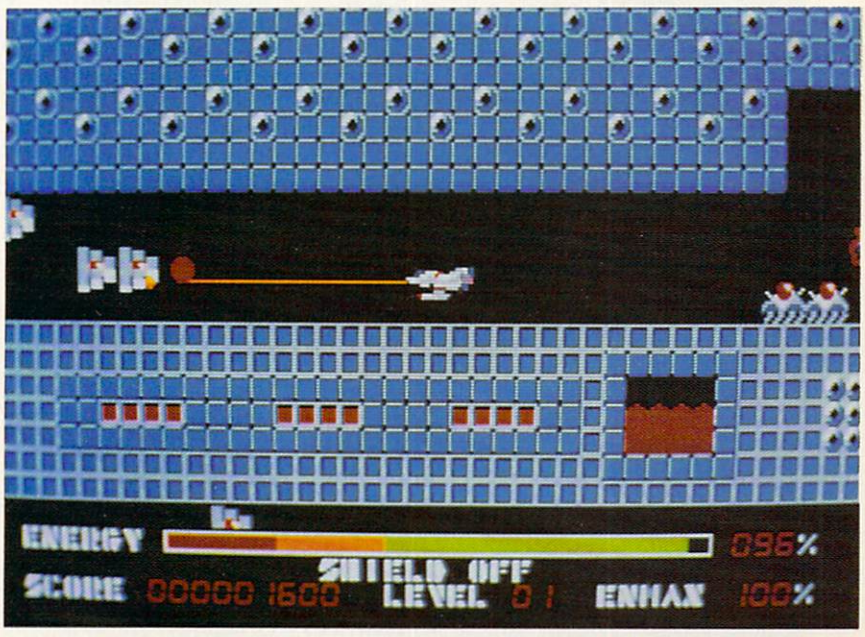
Touted as the most popular video game in Japan, THEXDER is a futuristic, arcade-type game that actually lives up to every promise made on the outside of the box: action, adventure, fantastic sound, and stunning graphics. Written by talented 68000 machine language programmers, THEXDER turns out to be a real screamer.

The Package

The package consists of a single disk and a very brief set of instructions covering how to use the numeric pad as controls, how to use a joystick, the meaning of the status indicators that will appear at the bottom of the screen, and your objective: To penetrate all sixteen levels of this heavily defended, erie underworld and destroy the central computer. To aid you in your mission, you've been given Thexder, a robot fighting machine complete with lasers, and the capability of transforming itself into a jet fighter. Transformation is accomplished by pulling down on the joystick or by pressing any of the bottom three numeric keys. However, you're warned that there's only one Thexder, and death on any level sends you right back to level one.

Ingeniously, when the instructions are unfolded, they form a wall-sized map of the first level of play, giving directions and a few hints. Around the border of the map are pictured some of the intriguing nasties you'll be facing. Some twenty in all are displayed, but there's a heck of a lot more than just these lurking out there.

Plug in that joystick!



On booting the copy protected disk, up comes a quick title slide displaying an alien landscape and the digitized voice of some young woman saying in Japanese, "Sierra On-Line presents THEXDER". Next comes the main title slide, listing copyrights and credits, while in the background plays the Moonlight Sonata by Ludwig van Beethoven. The game begins as soon as any key is pressed. Pressing the "J" key sends control to the joystick. If you have a joystick, I strongly suggest you use it!

When play starts, you find yourself literally up against a wall as the instructions warn, "There is no turning back!". Suddenly, you hear the same woman's voice proclaim, "Warning! Intruder!". There's nothing to do but turn, try to stay alive for as long as possible, and find you're way to the next level. The battle is on! Almost instantly you find yourself fighting off an army of the most imaginative and interesting critters to come along in quite a while. You'll face Tribars, Blademills, Rimquarts, and Clobters to name just

(continued on page 52)

VIP

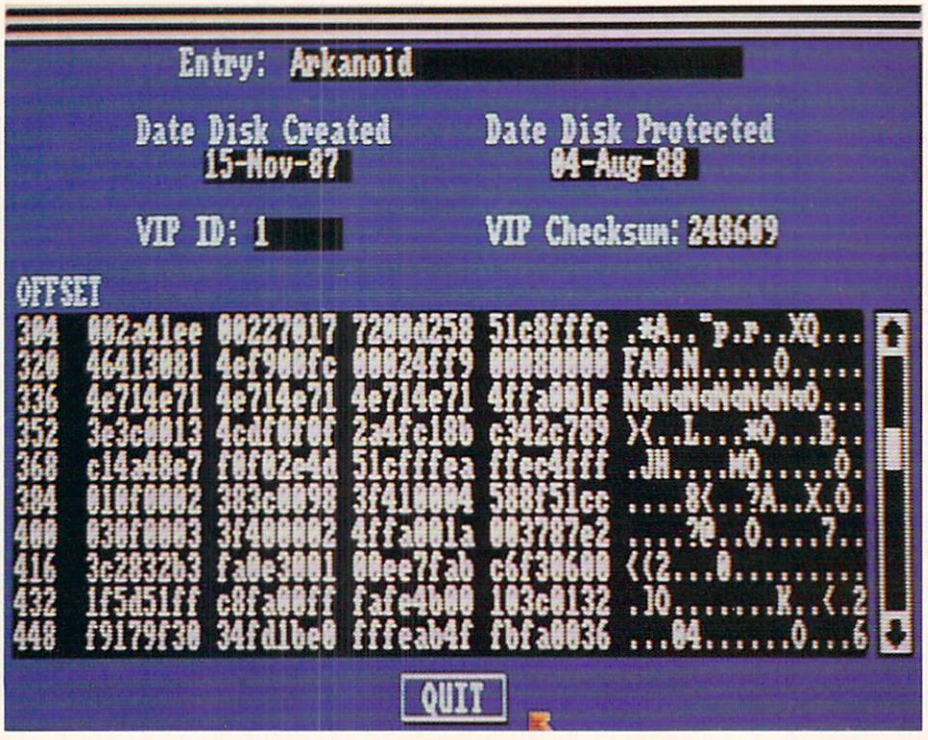
reviewed by Jeffery Scott Hall

Virus Infection Protection (V.I.P) from Discovery Software takes the first step in helping you protect your valuable software from the deadly disease. By acknowledging that this exists in the Amiga community, we are providing end users with not only information, but also a solution to the problem. Before we move on, let's further explore what exactly a computer virus is.

What makes a computer sick?

Did you know that your computer can catch a deadly and harmful disease? Well, while it might not be like you and me catching a virus it can certainly cause the computer to react violently. What I mean by this is that it can destroy your precious computer disks, any and all without discriminating. A virus will get inside your computer's RAM memory residing there until the computer is turned off.

While it's present, any disk that is inserted in the computer can be infected. However, this is only true for disks that are autobooting (i.e. Workbench, games, etc.). The way the virus gets from the computer's RAM to disk is that the virus itself will be copied onto the vital boot block sector, without which your disk will not function. When we are referring to a virus in essence we mean an "evil" computer program written by people wishing to derive pleasure from knowing that they've destroyed someone's valuable software. What is worse is that it's not just the Amiga community either, but rather all computer companies are finding themselves confronted with the same bad situations. For more information on computer viruses you should locate a copy of Time Magazine September 26, 1988 issue. Now that we've briefly discussed computer viruses, let's introduce the vaccination.



Checking for a virus with VIP

The cure for sick software

Virus Infection Protection will restore your damaged software to its original state by destroying any virus which is on the disk. Once the program is loaded, you will find five pull down menus which are: project, check, language, file, and viff entry. The project menu has four items on it which tell about the product, a brief description of the concept behind it, help with all the menu items, and quitting the program. On the check menu you'll find three items which are: examine, classify, and install.

The examine option allows you to insert a disk into any disk drive and check it for a possible virus. You will be happy to know that V.I.P. currently supports six of the most widely known and dangerous viruses which are: SCA, Byte Bandit, Byte Warrior, Obelisk, North Star, and Revenge. If your disk

contains a virus, a window will appear saying so along with the name of the virus which was found. The classify command allows you to enter new virus information into the database.

If you discover a disk to contain a virus which V.I.P. will not recognize, then you may select this command which will store the information so that from that point on it will know when it's encountered. Whenever one of your disks becomes infected with a virus, and it's standard AmigaDOS boot code, then you may use the install command to correct it.

The file menu has six items on it which are: new, load, save, save as, delete, and print. When first starting out, you should select the new option which will create a viff file. What this file will contain is all the boot blocks from the disks you wish to store, and can be given any name (i.e. If you want to store boot blocks for games, simply name the file games.) The viff file must be stored on an already formatted disk, don't use the V.I.P. disk for storage. The load option allows you to work with an already existing viff file, which may either be empty or contain boot blocks. Other options are self explanatory so I won't bother to describe them.

The viff entry menu contains seven items which are: view, make, remove, sort, verify, write, and print. You may view the boot code of any viff entry by selecting this option from the menu. The make command allows you to create a boot block from the disk of your choice to be stored in the viff database. For example, if you've checked Arkanoid for a virus and found nothing then you may wish to back it up in case disaster strikes. To do this, you simply insert Arkanoid into a disk drive and select make which will read the boot block so that it may be stored and recalled later.

The remove item allows you to delete any boot block which is stored in the viff database.

Sort allows you to organize your boot blocks according to date, alphabetically, date disk was created, and ID number.

The verify command allows you to compare the boot block in the viff database to the one which is on the disk.

If your disk gets a virus, you may destroy it providing you've already stored that boot block in the viff database. Using the write command allows you to select the appropriate boot block which will destroy any virus on the infected disk (i.e. If Arkanoid gets infected, select the Arkanoid boot block.) The print command allows you to get a hard copy of the actual data contained in a viff entry.

Summary

What makes V.I.P. so simple is its easy to use operating environment. All the user has to do is follow the manual and on screen instructions, which are designed very nicely. I think it's time that we all stood up and said, "Enough is enough and we're not going to take any more of this kind of abuse from people wanting to destroy the Amiga community!". That's just what I feel Discovery Software did by publishing such a valuable product as Virus Infection Protection. I commend them for keeping software users in mind, and for taking that first gigantic step in preventing valuable data loss.

I've been told that as more strings of viruses are found, they will keep updating V.I.P. to vaccinate them. In my opinion V.I.P. should stand for Very Important Product, which is exactly what it is and no user should be without one!

Virus Infection Protection (V.I.P.)
49.95

Discovery Software
163 Conduit Street
Annapolis, MD 21401
(301)-268-9877

•AC•

(***THEXDER**, continued from page 49)*

a few. Each is animated and has its own unique personality. Of course, there's the odd lava pit and acid lake to watch out for as well.

If you have time to look at the bottom of the screen, you'll see displayed your score, your energy level, the level of play you're currently on, and your Enmax: the highest amount of energy you're allowed to possess. There's also an indicator for shields status. Shields are activated by pressing the left Shift key on the Amiga keyboard, and can be a real lifesaver. However, try to use them sparingly. Activating your shields takes ten energy points, and completing a level of play without shields is rewarded by extra energy.

As you frantically scurry and fly through mazes, vast cargo bays, and caverns, you run into wave after wave of attackers, sometimes twenty and thirty at time. However, it doesn't take long to discover that the bad guys can be broken down into three groups: (1) Actively hostile: those determined to kick your metallic backside no matter what, (2) Passively hostile: these act as barriers and will not attack, but on contact will suck your energy dry in seconds, and (3) Semi-hostile: non-attacking, but again, lethal to the touch. However, destroying any of the third category of creatures gives you energy. Finding and destroying this third type of creature is essential for staying alive, as energy is the name of the game. Many of these energy creatures are cleverly hidden in the walls and must be blasted out. Fortunately, in robot mode, your lasers are heat seeking, and if your beams are suddenly attracted to a wall, it's a good bet that there's someone hiding there. Alternately, in fighter mode, your laser fire is always directed straight ahead, and with no heat seeking ability. So if you're getting low on energy, try to stay in robot mode as much as possible.

More than just your average shoot-'em-up, THEXDER has an added dimension. Many of the situations you find yourself in are actually clever, logic puzzles, where the solutions may consist of finding ways of getting free energy, or suckering out the nasties by some special means. This helps keep repeated play from becoming boring, and motivates you to try a new strategy on the thing that wasted you the last time out. At the same time, THEXDER is so rich in complex, little details that you're almost guaranteed to notice something new each time you play. Praises go to the authors, S. Uesaka and G. Godai!

Summary

As for the overall game; as a "quarter saver", I found it flawless. It's visually stunning, the animation is as smooth as silk, all of the creatures are interesting and clever, and the forbidden underworld is exceedingly intricate and full of surprises. Obviously written for 16 bit machines, this game does real justice to the capabilities of your Amiga. If you like action/arcade games, THEXDER is definitely worth the money. Check it out, and I'll see you on level eight.

THEXDER
Sierra On-Line, Inc.
P.O. Box 485
Coarsegold, CA 93614
(209) 683-4468

•AC•

ROOMERS

by the Bandito

[The statements and projections presented in "Roomers" are rumors in the purest sense. The bits of information are gathered by a third party source from whispers inside the industry. At press time, they remain unconfirmed and are printed for entertainment value only. Accordingly, the staff and associates of Amazing Computing™ *cannot be held responsible for the reports made in this column.]*

Aegis is looking hale and hearty these days, thanks to a healthy infusion of capital (more than a million, it's said). Their booth at AmiEXPO was impressive, displaying their new confidence. The word is that their cash flow crisis is solved, new people are being hired to replace the old guard that left, and product development is proceeding apace. See how rapidly things can change in this business? It just goes to show you. (Remember what Mark Twain said about his premature obituary?)

Aegis's new *Audiomaster II* sound editing program looks like a winner: it turns your Amiga into an oscilloscope and gives you real-time effects like echoing—speak into the mike and your voice is processed through the Amiga in real time. Very impressive. Aegis also announced a new program called *Videola*—designed for editing ANIM files; and a new version of *Sonix* that offers more editing features and full MIDI compatibility.

Also at AmiEXPO, Allen Hasting's new mini-movie Rush Hour was a hit, drawing crowds at every showing at the NewTek booth. No new movies for a while, since he's at work programming on a top secret project with NewTek. (The Bandito's informant proved extraordinarily fragile during routine questioning, so no details yet. They just don't make informants like they used to.)

Possibly the strangest thing at AmiEXPO was the lack of new products. Oh, there were plenty of games, and some new versions of old products were announced or hinted at, but nothing to make you jump up and run down to your local Amiga dealer. Most companies were showing the same old things in the same old way. Seems like the turmoil at many Amiga developers has slowed down product development a bit, and most developers are putting their effort into marketing their current product for Christmas rather than introducing new product. Look for a slew of new product introductions in the first quarter of next year, including some long-awaited hardware and software.

AmiEXPO also had smaller crowds than expected. The problem could be related to the prices: $20 for one day admission, $15 to park at the hotel parking lot, plus the aggravation of having to deal with LA traffic (which is abysmal even on the weekends). So throw in a little bit of money for food, and you're talking about nearly fifty clams just to pick up a bagful of flyers and see a Madonna imitator. Some developers were grumbling that it's not worth it unless there's a better turnout. It will be interesting to see how this compares to the upcoming World of Commodore show in Philadelphia.

Is Mediagenic (actually, their Activision division) negotiating to buy one of their affiliated labels that's made a name for itself in the Amiga market? Well, it certainly sounds like a good match-up. Stay tuned to this space for more information.

The Bandito mentioned *Dragon's Lair* before, and here's an update. The game comes on six disks, retails for $60, and they're distributing a demo that has about six to ten seconds of really superb video on one disk. But cynical observers note that the original game had about 20 minutes of video. Six disks at 10 seconds per disk yields about 1 minute of game. Either they've got some incredible compression, or the whole game doesn't look quite like the demo we've seen. What will the game really be like?

New Technologies Department: A new CD format has been created — CD+MIDI, which stores up to 16 channels of MIDI data (along with graphics) in the subcode area of the disc. Making a CD player compatible with CD-MIDI adds about $2 to the manufacturing cost, so expect to see a number of the new players with this capability. CD-MIDI lets you relocate instruments or voices in the playback mix, change the sound of the arrangements by revoicing parts with your own MIDI equipment, and change the tempo or key. You can even print out sheet music, resequence, or repeat sections if you've got a computer hooked up. Makes an interesting addition to an Amiga music studio.

In other news, an interesting conference on computer game design just took place in California. It was described by attendees as similar to Electronic Arts' Artist Conference, only independent of any software developer. It was put on by Chris Crawford, the grand old man (or *enfant terrible*, depending on your point of view) of computer game design. There were seminars for game design, technical issues, and legal stuff to do with computer games. Mainly, of course, people were cutting deals in the hallways and exchanging juicy gossip.

The Amiga seems to be commanding more respect as a target machine for games that can make money and generate great PR value, though the first targets for development are usually MS-DOS machines and the C64. The developers were near unanimous in their

(continued)

hatred for the marketing people at almost every publisher, regaling each other with stories of the ignorance and cupidity of market-droids. According to the developers, it seems that many of the marketing people at the biggest publishers come from strange backgrounds, like selling dog food, shampoo or fertilizer. Many of these market-droids are not only ignorant of computers and games, but they don't WANT to learn anything, which is really annoying. At least the smaller publishers don't have the problem (well, maybe not to the same degree).

Lotus is planning to do 1-2-3 for the Macintosh, IBM mainframes, and other odd types of computers. Could they be looking at WordPerfect's success, and seriously considering an Amiga version? The market's getting big enough. If they aren't, they should be.

Commodore is spreading its advertising dollars a little further afield these days. New ads, appealing to the professional graphic artist, are appearing in *Advertising Age* and *Print* magazines. There's a very nice Amiga ad in Mix magazine, the recording industry rag. It's a two page spread with very slick graphics and text. It looks like Commodore is getting serious about going after the music market. They're signing up many music dealers, especially targeting the ones that Atari acquired during their big push into the music market. Unfortunately for Atari, their success has paved the way for Commodore to do the same thing.

Overall, Commodore's doing great, showing better profit numbers every quarter. The Amiga is accounting for a bigger percentage of their sales with each passing quarter, which is just fine since it's the most profitable of all the computers they sell (the A2000 is especially profitable). Commodore's great fear, that C64 sales would collapse before the Amiga has a chance to pick up the slack, has proven unfounded as yet. While C64 sales continue to slow, there's no sign of the bottom falling out of the market. Commodore's busy picking up new retail accounts for the Amiga: Computerland will carry the A2000 in some stores; Entre computers carries it; and Fry's, the hacker's computer store in Silicon Valley, has dumped the Atari ST and is carrying the Amiga.

The A500 is developing as the premier game machine, and that's what's driving sales. The A2000 has a tougher road to travel. It needs something to really break into the business market, something like the LaserWriter/PageMaker combination that made the Macintosh sell. The Bandito thinks the answer lies somewhere in the desktop video market...

Look for some good deals on Amiga 2000s this Christmas. The price may not be lower, but dealers will probably bundle software and hardware. This is a result of Commodore giving them better profit margins in an attempt to boost A2000 sales. Don't look for any price decreases for the foreseeable future; Commodore will likely use rebates, bundling, or other promotions rather than reduce the list price.

There's a new problem for Amiga dealers — finding shelf space for all of the Amiga software titles. Many dealers are having to rearrange their shelves, or at least think more about what to stock. At last count there were about 500 different Amiga titles available (about half of them are games). This portends some big changes in the Amiga software market. Gone are the days when any software title could find a spot on the shelf. It's becoming dog-eat-dog out there, and the days of the ziploc bag, wrote-it-in-my-attic-in-my-spare-time software are just about over. With every passing day, the cost of bringing a piece of Amiga software to market (with distribution, packaging, and advertising) increases. Which probably means we'll see a little less innovation *(sigh)* as the products are brought out by larger and larger companies.

Don't be surprised you see one or two major Amiga developers pulling back on their advertising. There's a little bit of the cash-flow flu going around, and some developers have a bad case. Not to worry, they'll feel better after Christmas, when Santa brings them hearty sales figures for their stocking.

The ST is fading fast — developers no longer even discuss doing a version of a successful IBM or Amiga product for the ST, since there's no money in it. So many of the staunch Atari developers are looking other places for revenue. Some are trying to crack the IBM market, and others are trying to take their 68000 expertise to the Macintosh and Amiga markets. Antic, one of the ST's most fervent supporters and a longtime Amiga-basher — look at some back issues of the magazine—has now deciphered the text font on the wall and started Amiga software development. They announced that *Zoetrope*, Jim Kent's animation program for the Amiga, will be available soon. They're also selling *Pioneer Plague*, billed as the first all HAM mode action game. They're very careful not to mention the word "Atari" when they talk to Amiga fans...

David Klein, Electronic Arts' superstar salesman, has left to become Commodore's new Vice-President of Marketing. Those in the know say that he should do a whiz-bang job. Look for some new and different approaches to Commodore's marketing and distribution next year. By the way, Klein's specialty at EA was selling to mass-market accounts like Toys R Us.

Deep C, the Bandito's Commodore connection, reports several interesting developments in the Commodore laboratories. The first one is a better blitter chip, more than twice as fast as the current model. And of course they're working on 15-bit color (that's a palette of 32,000 instead of 4,096), with a new HAM mode that allows all of those colors to be displayed at once. (This ties in with something the Bandito heard elsewhere, about a developer who's working on applications that work in many more than 4096 colors on the Amiga.)

But the hottest idea being bandied about at Commodore is this: take an Amiga 500, remove some of the unnecessary stuff like the keyboard, the disk drive, the mouse, and most of the ports, put in a cartridge port, and what do you have? A Nintendo-Killer, if you can price it around $200. That's right, Commodore is thinking about taking a shot at the videogame market. The big discussion centers around how much memory to build in — some reactionaries are arguing for 256K, but the progressives want 1 megabyte. Bet on at least 512K so they can run the majority of current Amiga games. If this happens, there'll be a tremendous opportunity to make money from the current crop of Amiga games, so you can bet that developers are watching with great interest.

While we're on the subject of videogames, let's talk a bit more about CD-I. Adding to the confusion surrounding this new hardware is RCA's Digital Video Interactive (DV-I), which is similar to CD-I except that it offers full-screen, full-motion video (by using some sneaky compression techniques). RCA initially demoed DV-I at a CD-ROM conference and completely upstaged CD-I. After-

wards, RCA said that they were just presenting the technology and didn't know what kind of product it would make. The consensus at that time was that DV-I would be just a high-end business hardware option, since RCA's first guess was that a DV-I board for a PC AT computer would cost about $3000, putting it well out of reach for any but the most esoteric applications. But now Intel (makers of the 80x86 chips) has just bought the rights to the DV-I chip set and the technology, so we could see DV-I as a consumer device after all. Or perhaps as a computer industry standard, extending the CD-ROM format. But don't expect anything until the 1990's.

There's a flood of games coming from overseas, and the Bandito has noticed that each has its own distinctive packaging — from CD jewel boxes to albums to boxes to baggies. Some of the "artwork" (to use the term loosely) would gag a maggot. (But then, a few packages from our side of the Big Ditch look like old road kills, themselves.) While the graphics and sound are usually pretty good, most of the imports are lacking when it comes to game design. The Bandito wishes there was somebody reliable out there who could review ALL of them as they come out, so that people wouldn't waste $30 on trashy games (and add to the trade deficit, besides).

Yet another flight simulator, Skychase, is coming out, and this one boasts the fastest frame rate of any flightsim on the market (it's wire-frame instead of filled solids). You can fly all sorts of planes, even a paper one, in head to head competition with another player or the computer.

Dale Luck (one of the original Amiga development team who's still hanging in there at Amiga Los Gatos) has developed a three-button optical mouse for the Amiga. It's part of his X-Windows development effort; on sale in January, or thereabouts. The mouse is twice as accurate as the ordinary mouse, and less prone to mechanical trouble.

New Hardware Dept.: The NeXT computer was introduced October 12 to great hullabaloo. It does have some neat gimmicks (a 256 megabyte erasable optical disk, for one), but it won't even be available in quantity until the middle of next year, and then only to universities (and at a hefty $6500 price tag for a monochrome machine). Of course, they'll probably start selling it to everybody in 1990, but by that time the Amiga will have much of NeXT's capability... won't it? Hello, Commodore?

Speaking Of Other Guy's Hardware Dept.: According to inside sources at Apple, IIGS sales are disappointing, and the recent price hike won't help matters any. Apple seems to think the Amiga is one reason — which may be why dealers are reportedly being pressured NOT carry Amigas if they carry Apple products. And don't look for the IIGS Plus real soon; it now looks like they won't bring it out until the summertime, or maybe even the fall. They're busy concentrating on Macintosh sales.

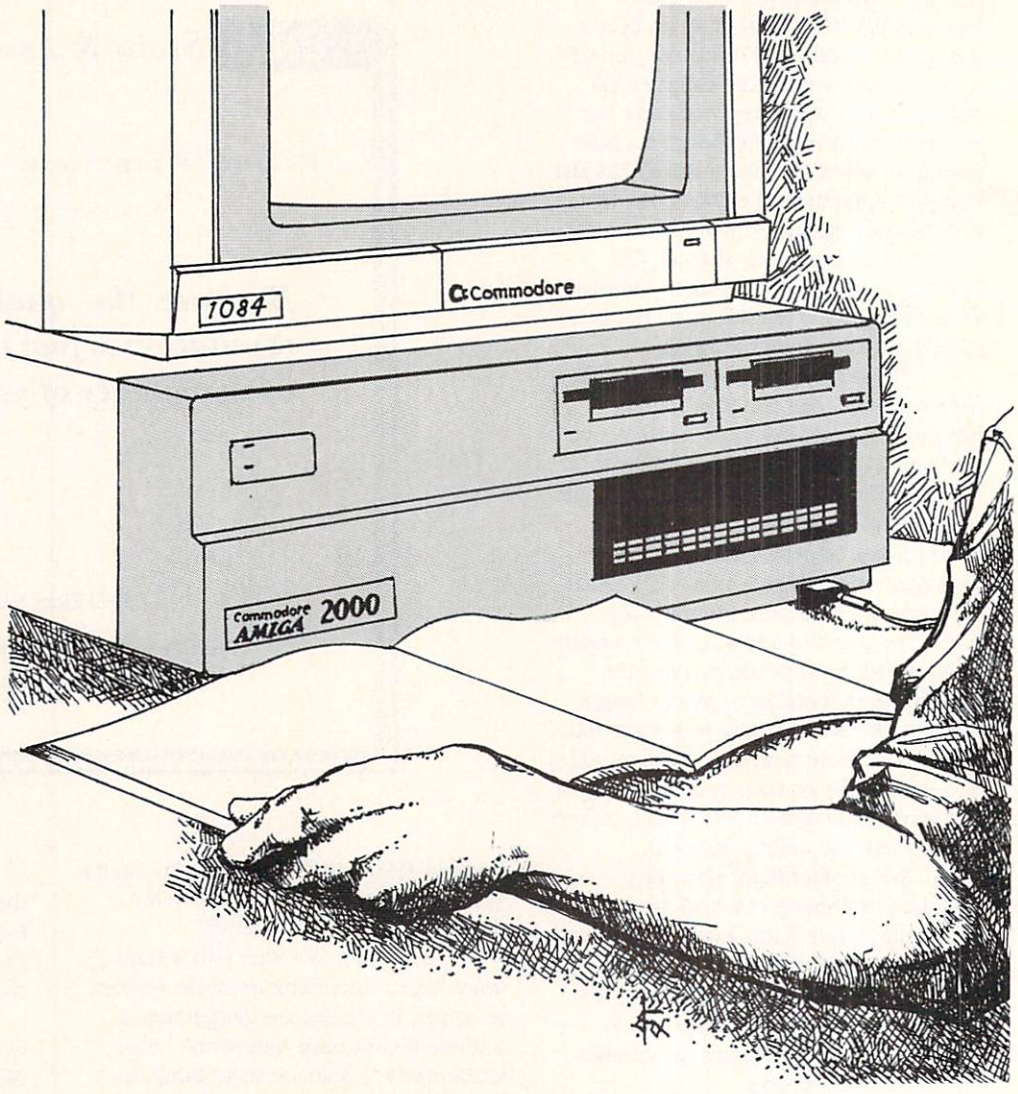
One thing to keep your eye on — will Apple drop the Mac Plus from the product line, or drop the price and keep it around? Things would be easier for the Amiga if it didn't have to compete with a $995 Macintosh. Sure, technologically it's not much of a contest, but remember Apple's marketing muscle. Let's hope the plug gets pulled on the Mac Plus.

There is no truth whatsoever to the rumor that a major Amiga developer is porting its animation software to the Etch-A-Sketch, even if the interfaces are similar.

More Amigas in the media: they're in the John Carpenter film *Prince of Darkness* (even using Amiga text fonts and wire-frame animation); they're generating some of the graphics for the Max Headroom Coke commercials; they're being used by the good guys in the new *War of the Worlds* TV series; they're helping out with the production of *Who Framed Roger Rabbit.* A few more hit films, and maybe we'll see good ol' Ami written up in Hollywood gossip columns: "Ami was seen out on the town partying hearty with Ashton Tate, spurning her recent boyfriend Mike O. Soft..." From there, it's only a short step to the Betty Ford Clinic to deDOSify...

•AC•

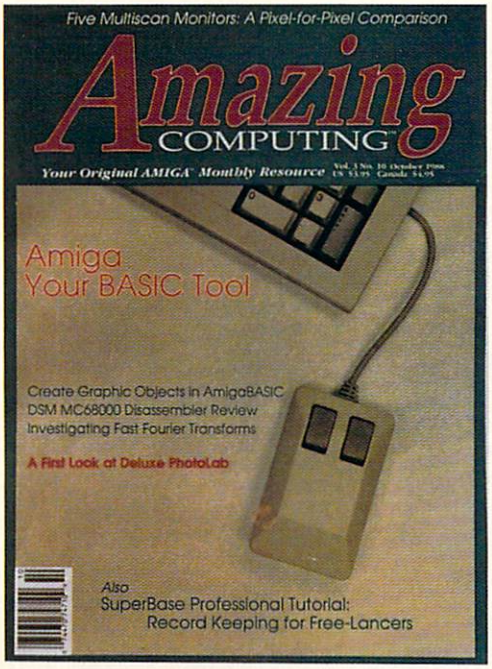
For the Amiga user who has everything...



What do you give the Amiga user who has everything? Well, there is one thing that *every* Amiga user desperately needs—solid, useful, practical information. And you know there's only one source for that kind of information—your Amiga monthly resource, Amazing Computing.

Each month, Amazing Computing connects creative, ambitious users to the power behind the Amiga. The Amiga has the potential to perform in endless computing arenas. Amazing Computing provides the insight, detail, and examples you need to make the Amiga fit your needs and your dreams.

An AC Gift Subscription!



Ho, Ho, Ho!

AC goes beyond magazine staples such as product reviews, trade show reports, and interviews to get down to the core of users' needs. Sure AC publishes unbiased, highly-regarded reviews and tasty industry whispers, but we really concentrate on hands-on applications. Step-by-step hardware projects, desktop video applications, down-and dirty-tutorials, and coverage of all levels of programming keep Amiga users reading AC each and every month. And for fun-lovers, AC also provides thorough coverage of those great Amiga games and other entertainment options.

Want to make an Amiga user happy year round? This year, give your Amiga enthusiast a priceless gift: solid, useful Amiga information. That rare commodity, information you can really put to work, can be found in only one source—your Amiga monthly resource, Amazing Computing.

PD Serendipity

Insight into the World of Freely Redistributable Software for the Amiga™

by C.W. Flatte

In this episode of PD Serendipity, we'll cover Fred Fish 155 - 161. Let's get started.

Fred Fish 155

AsmExamples
A couple of assembly code examples by Henrik Clausen. Executables are useful but the code is of considerable benefit to beginning assembly language programmers.

Bison
(Update to Fred Fish 136)
A replacement for unix "yacc" command (Yet Another Compiler Compiler). This is from the GNU (GNU is Not Unix) effort. Contains updates to the version on disk number 136, submitted by two separate sources. Includes the entire source. The programmers involved are Bob Corbett and Richard Stallman, updates by William Loftus and Scott Henry

NoSmoking
This is a sample program showing the use of a recoverable alert while displaying a personal health message. Includes source by Theo Kermanidis

Scenery
A very nice assembly language random scenery generator. Generates very realistic looking landscapes. Includes intuition interface and lots of menu options. Version 1.0, binary only. By Brett Casebolt

Fred Fish 156

Blocks2
Amusing and colorful display of a moving trail of "blocks". Update to version on disk number 71, however this version also includes source. By Gary Walker

Flex
Flex is a replacement for the UNIX "lex" (lexical analyzer generator) program that is faster than lex, and freely redistributable. Includes source. Authors: Jef Poskanzer, Vern Paxson, et. al. Submissions by William Loftus and Scott Henry

Go64
Another screen hack aimed at an earlier Commodore product (Not to be confused with the commercial product Go-64! from Software Insight Systems). Includes source. By Joerg Anslik

Grammars
A group of lexical grammar files for Ada, C and Pascal for use in conjunction with the flex program on this disk and the bison program on disk #155 Authors: Various, submitted by William Loftus

OOPS!
Tired of the monochrome background color of your Workbench or CLI? Then try this colorful screen hack to brighten things up! Includes source. By Joerg Anslik

Fred Fish 157

60or80
A small utility to toggle the 60/80 column text modes without having to go through preferences. Works from either the CLI or the Workbench. Includes source. By Mark Schretlen

AmicForm
Creates a phonebook containing only those areacodes and exchanges reachable through PC-Pursuit. Input any of Chet Solace's Finalist BBS lists and it creates the phonebook in a form usable by AmicTerm and a number of other popular terminal programs. Version 1.3, Binary only. By John Motsinger

AnimBalls
A nifty little animation program that allows you to create a collection of balls in three-space and then interactively rotate them in real time using the mouse. Includes source. By Jim Guilford

BootBack
A handy little utility to copy and save the boot block from a disk, then later restore it should the disk get stomped on by some ugly virus. Includes source. By David Joiner

ECPM
A CP/M emulator for the Amiga. Emulates an 8080 along with H19 terminal emulation. Update from version on disk number 109. Includes source. Author: Jim Cathey; Amiga port by Charlie Gibbs; Significant improvements by Willi Kusche

(continued)

TxEd *Plus*™

The Text Editor for the Amiga™, *Plus* a whole lot more.

BlitzDisk	Disk cache, speeds up floppy and hard disk reads up to 2000%.
FastFonts	Speeds up text display.
FunKeys	Hotkey window manipulator.
ARP	Latest versions of the AmigaDOS™ Replacement Programs.
AREXX	Demo version of the AREXX, the macro processor used by TxEd Plus.

TxEd *Plus* is a powerful yet simple text editing environment designed to meet all of your text editing needs. It's small, fast, and fully configurable. The AREXX macro language connection makes TxEd *Plus* an ideal foundation for your complete multitasking system.

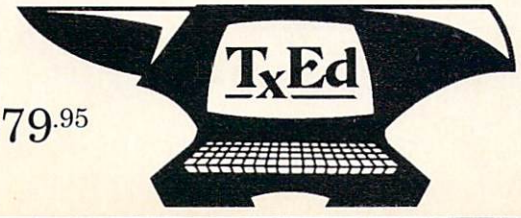
"As a programming and general-purpose text editor, TxEd Plus sets the standard."
-Warren Block, Sept '88 INFO Magazine

"(TxEd V1.3 is) ...a very good editor and an excellent value."
-Jan&Cliff Kent, Vol. 1 #9 Amazing Computing

"FastFonts/BlitzDisk provides much more than Facc II for a similar price."
-Warren Block, July '88 INFO Magazine

"Yes, get TxEd if you're editing almost anything on the Amiga."
-Bruce Webster, July '86 BYTE Magazine

"I found the ARP commands to be smaller, faster, and more powerful than their BCPL forerunners."
-Jeff Blume, July '88 AmigaWorld



$79.95

Microsmiths, Inc
PO Box 561, Cambridge MA 02140
(617) 354-1224 BIX: cheath CIS: 76004,1766

MC and Visa accepted. Mass Residents add 5% sales tax.
Amiga and AmigaDOS are trademarks of Commodore-Amiga, Inc

KeyFiler
A BBS message file sorter that allows sorting by keyword. Includes a textreader, Soundex matching, and limited wildcard capabilities. Version 1.0, Binary only. By John Motsinger

ScreenZap
A little utility to clean away screens that are left by ill-behaving programs. It will kill every screen behind the WorkBench, noting how many it gets. The screens in front of WB are not affected. Includes source. By Lars Clausen

SetPrefs
Allows you to build a whole library of preference settings and instantly switch back and forth between them. Affects all preference settings not just the colors. Very useful for machines with multiple users or multiple external devices. Includes Amiga's default and various sample preference settings. Binary only. By Martin Hippele

Xicon
Xicon lets you use icons to call up scripts containing CLI commands. This is version 2.01, an update to the version on disk 102. Includes source. By Pete Goodeve

Fred Fish 158

DiskX
Nicely done Sector-based disk editor. Binary only By Steve Tibbett

MemBoardTest
Originally designed for production testing of A1000 memory boards. Very nice intuition interface. Version 2.4, Includes source in Modula. By George Vokalek

MSDOS
A program to list files written in standard MS-Dos or Atari ST format. The files can then be copied to Ram and rewritten to disk in Amiga-Dos format. Binary only, Shareware, Version 0.1. By Frank Wubbeling

PCBTool
An early version of a shareware PC Board layout program Lots of options including variable size pads and traces, grids, grid snap, layers, zoom, selectable centering, text and more. This version does not support printer/ plotter dumps or libraries. Version 2.6, binary only. By George Vokalek

ScreenX 2.1
A handy little background utility that provides a small clock/memory counter in its inactive mode and a versatile screen manipulator when called upon. Binary only with source available from author, Version 2.1. By Steve Tibbett

TaskX 2.0
A "real-time" task editor. Lets you list and set the priorities of all the currently running tasks. Binary only, Version 2.0. By Steve Tibbett

VirusX
Update to the version on disk number 154, checks for a couple of additional new strains. Includes source, Version 1.6. By Steve Tibbett

YachtC3
Update to the Yachtc program on disk #10, contains some fixes and incorporates a simple sound process. Version 3, includes source. Author: Sheldon Leemon, with enhancements by Mark Schretlen

Fred Fish 159

Free
A little command to put in your c directory that returns memory status and number of tasks currently served by EXEC. Includes source. By Joerg Anslik

MidiTools
A group of several different utility programs for those who run a Midi system. Binary only. By Jack Deckard

StarChart
Nicely done intuition based program to display and identify about 600 stars, galaxies and nebulae visible in the Northern hemisphere. Version 1.2, includes source. By Ray R. Larson

TaskControl
Nicely done task-handling program allowing you to put to sleep, kill or change priorities of the all the currently loaded tasks. Also potentially GURU-producing, so be careful what tasks you kill, change priorities of, etc. Handy window sizer will reduce it almost to an icon to hang around until you want to use it. Binary only. By J. Martin Hippele

TUC
"The Ultimate Clock". Another window title clock/memory minder. This one is in 132 columns! Also gives the free memory on drives DF0, DF1 & DF2. Includes source. By Joerg Anslik

Fred Fish 160

Calls
A little utility to help analyze the flow of a C-program by laying out the functions called in a hierarchical manner. Author: Originally from Usenet with major revisions by Kevin Braunsdorf, Amiga port by George MacDonald

Check
A useful little utility for finding structural errors in C-source code. Many command-line options. Version 1.03, binary only. By Keith Elbertson

Dis
A 68000 disassembler, written in assembly, this is an update to the version on disk #128. Includes source. Author: Greg Lee with enhancements by Willi Kusche

DMouse
A versatile screen & mouse blanker, auto window activator, mouse accelerator, popcli, pop window to front, push window to back, etc, widget. Version 1.09, includes source. Update to version on disk number 145. By Matt Dillon

DWIP
"Daisy Wheel IFF Printer". A graphics printing utility that allows the printing of IFF pictures on a daisy wheel printer. Includes source. By Ken Van Camp

M4
A UNIX M4 look-alike macro processor intended as a front end for Ratfor, Pascal, and other languages that do not have a built-in macro processing capability. Pd M4 reads standard input, the processed text is written on the standard output. By Ozan S. Yigit (oz)

MemoPad
A shareware intuition-based memo reminder program. Nicely done. Update to version on disk #146, version 1.2, binary only. By Michael Griebling

NeuralNets
A neural network example using the generalized back-propagation delta rule for learning, specifically applied to the tabula rasa Little Red Riding Hood instance. By Josiah C. Hoskins

Fred Fish 161

Friends
Cute little screen hack with command-line options to keep your mouse pointer company when you step away. Includes source. By Michael Warner

Getsprite
A simple little program to convert Dpaint brushes into C-source. Binary only. By Michael Warner

IncRev
A handy little program that will automatically increment the revision number of a program every time it is recompiled.Binary only. By Bryan Ford

LGZ 0.1
A Map generator/editor for the LGZ game. Not extremely useful if you don't happen to play that game, but good source example of intuition interfacing. Vesion 0.1. By Lars and Henrik Clausen

Mackie
A versatile cli/macro-key initiator based on POPCLI with a unique method of "screen-blanking". I won't say more, just try it! Version 1.1, includes source. By Thomas Rokicki

Nag
A shareware appointment calendar with it's own editor and a unique 'nagging' feature utilizing the Amiga's voice and audio devices.Version 1.6, binary only. By Richard Lee Stockton

Perl
Practical Extraction and Report Language, an interpreted language optimized for scanning arbitrary text files, extracting information from those text files, and printing reports based on that information. By Larry Wall

VRTest 3.2
Another anti-virus utility that allows visual inspection of ram starting a $7E7FE, ram cleaning, bootblock inspection and vector monitoring/ reseting. Written entirely in assembly language. Version 3.2, binary only. By Babar Khan

XBoot
A very simple utility to convert a boot block into an executable file so you can use your favorite debugger (Wack, Dis, etc.) to study it. Includes source. By Francois Rouaix

That's all that I have time for!
Until next time...

Gotcha!

—C.W. Flatte

Send your PDS Questions to:

C.W. Flatte
c/o *Amazing Computing*
P.O. Box 869
Fall River, MA 02722

•AC•

1> by Rich Falconburg

The Command Line

The Continuing Guide to the CLI

Although AmigaDOS is a powerful operating system with some very useful commands and a fair amount of flexibility, the creators left a few things out. Fortunately, the Amiga has attracted a lot of attention from top notch programmers that have helped to fill in these gaps. These developers occasionally write small programs to fill a specific need that arises during program development. To our great fortune, many place the utility program in the Public Domain.

These programs are free for the downloading or copying time you wish to invest. Some of the authors request a modest donation if you like the program and use it. Others are classified as "Shareware" meaning that the program is free for you to use to determine if it fills your need. If you use it, you are asked to send a payment to the author ranging from $10 to $25. Quite reasonable. Particularly if you consider that many of the programmers will reciprocate by sending you the latest version of the program, complete documentation, and often, additional sample programs of other utilities offered.

AmigaDOS don't fail me now!

When the commands of AmigaDOS fail us, we can often turn to the wide range of "freely distributable programs" that exist in numerous User Groups and on hundreds of Bulletin Board Systems throughout the country. In the next few issues, I will examine some of the programs that have been written to address several weak spots in AmigaDOS. If you already own a modem, you have at your disposal a door to a fantastic world offering something for everyone. If you don't own a modem, I strongly urge you to put one on your list of "Toys I MUST have".

User Groups

A User Group is the next best thing. You will find people of varying ages and backgrounds with a dedication to your favorite machine that could be classified as "fanatical". Ask some of the "Old Timers" what utility programs they find most useful. There are hundreds of titles in the Public Domain and finding just the right program to fit your needs can often be discouraging. The list provided at the end of this magazine, The AMICUS & Fred Fish Public Domain Software Library, is a resource of unparalled distinction. Use it.

ConMan

The line editor used in the console windows must have been written in the dark ages. It's a real pain to retype an entire line that is rejected because of a typographical error. Several solutions to this problem were introduced early on with each having various inconsistencies of their own. Then along comes William Hawes with ConMan and the sun begins to shine.

The keyboard communicates to the Amiga through a type of driver that interprets the key codes and displays the results. The one supplied by Commodore has a single distinguishing editing feature: the BACKSPACE key. How can you build the greatest computer of the decade and then curse it with a keyboard interface seemingly derived from the Teletype? We may never know. ConMan up-roots the default console handler and replaces it with one that provides powerful editing capabilities. Once ConMan is installed, every console window opened will use the new console handler. So what does ConMan do?

A. **Editing**

1. Complete editing facilities using the Left and Right cursor keys.

2. The DELETE key works as it should. Delete words backward (F7) and forward (F8). Delete from the cursor to the end of the line (Control Y) or to the start of the line (Control U) or delete the entire line (Control X). Delete all lines, including type ahead lines (Control Z) or return them (Control R).

3. Skip to next word backwards (Shift Left Arrow) and forward (Shift Right Arrow).

4. Toggle between Insert and Overstrike (Control A) or force Insert mode (Control ^).

5. Support of Xon (Control S) and Xoff (Control Q).

B. **Command History**

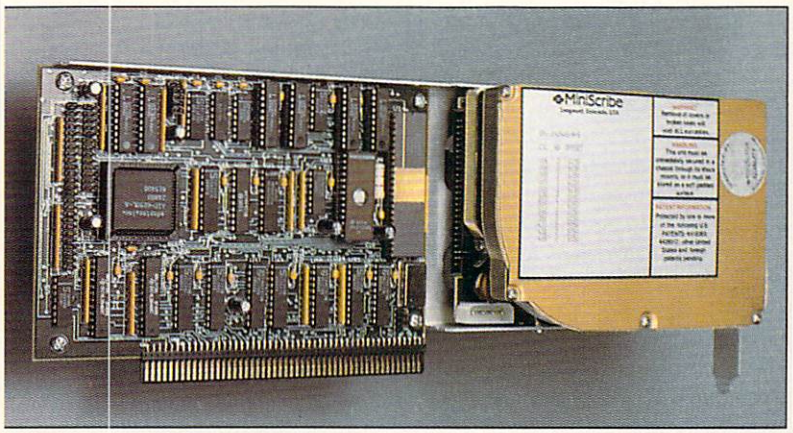
1. Recall previously entered lines by pressing the Up or Down Arrow keys. Shifting each of these will recall the latest and oldest lines respectively.

2. Clear the history buffer with Control B.

3. Buffer size and recall mode options that may be set when ConMan is first started. This affects all console windows. If you need to change these later, the utility program SetCMan makes it easy. The default buffer size is large enough to hold 25 to 30 lines.

(continued)

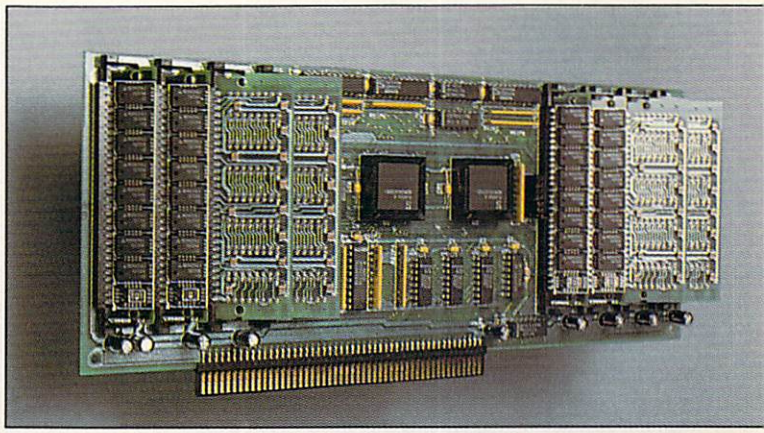
HardFrame/2000
The Super-speed, DMA, SCSI Hard Disk Interface with 1.3 Autobooting



How fast is fast? **HardFrame/2000** transfers data at Amiga bus speeds! It's actually faster than the hard disk mechanism itself! And even more important in the Amiga's multitasking environment, **HardFrame/2000** has extremely efficient DMA circuitry to get on and off the bus in almost no time at all: 280ns to get on; 200ns to get off. **HardFrame/2000** autoboots under AmigaDOS™ 1.3 and is fully compatible with the new Fast File System. The core of any DMA SCSI interface is its SCSI protocol chip and DMA chip. MicroBotics has chosen the new, high performance Adaptec AIC-6250 SCSI chip, capable of up to 5 megabytes per second raw transfer speed, and the Signetics 68430 DMA chip running at 12.5 megahertz. Then we added additional FIFO buffering and enabled 16-bit wide data transfers for maximum throughput. The sophisticated design of **HardFrame/2000** provides for automatic SCSI arbitration, selection and reselection. The hardware supports either synchronous or asynchronous data transfer. **HardFrame/2000** can function as either the SCSI bus initiator or the target and can reside in a multiple master environment. Physically, **HardFrame/2000** is optimally flexible: the compact, half-size card comes attached to a full length, plated aluminum frame. The frame has mounting holes positioned to accept standard, 3.5" SCSI hard disk units such as those manufactured by MiniScribe, Seagate, Rodime, and others (hard disk mechanisms must be supplied by the user or his dealer as a separate purchase item). Alternatively, you can cable-connect to a SCSI drive mounted in your Amiga's disk bay or in an external chassis. As many as seven hard disks may be connected to a single HardFrame. There is no size limit on each disk. **HardFrame/2000** includes a 50-pin SCSI cable and header connectors for either 50-pin or 25-pin cable connection. Also included is a current tap to power frame-mounted drives directly from the slot itself. **HardFrame/2000** comes complete with driver, installation, and diagnostic software. Available September 1988. Suggested list price, $329 (hard disk not included).

The HardFrame/2000 photo shows the product with a MiniScribe 20 megabyte hard disk installed. Hard disks are not included in the purchase price of HardFrame. Note that if placed in the first slot, HardFrame uses only one slot.

8-UP! The Eight Megabyte Memory Card with Amiga™-specific DRAM Controller Logic



All the memory space you and your Amiga 2000 need –in a modern, highly integrated FastRAM expansion board. In **8-UP!**, MicroBotics went all the way to provide you with a truly Amiga-specific memory design to meet the special demands of the Amiga's high speed multitasking environment: The heart of any memory expansion is its *DRAM controller circuitry*. Rather than compromising with off-the-shelf parts, MicroBotics developed its own, custom controller design and built it into high-speed, *Programmable Macro Logic* chips (Signetics PLHS501). These new, super chips (each **8-UP!** uses two PML's) permit MicroBotics to employ *sparse refresh* technology to assure that your **8-UP!** is a truly zero wait-state/minimal-refresh-collision memory design. If you're putting eight megabytes in only one slot, that means that you probably have plans for your other A2000 slots. **8-UP!** gives you new freedom to do that planning since, unlike other ram peripherals, it is an extremely low-power memory card– a single, fully-loaded, 8-megabyte **8-UP!** draws an astoundingly efficient 0800 milliamps! That's less than *two-fifths* of the power "budget" for a single slot! Low power draw also means that the card is cool-running for reliability and long life (not to mention a cooler Amiga!). **8-UP!** offers you maximum flexibility in memory configuration: it is organized into two separate *PIC's* (Amiga-speak for autoconfiguring peripherals). Each **8-UP!** PIC consists of four SIMM module sockets; these sockets accept either 256k-byte or 1 megabyte *SIMM's* (Single Inline Memory Modules). You can also purchase optional *PopSIMM* boards from MicroBotics; fill them with conventional RAM ; then use PopSIMM's to fill your **8-UP!** The card can run with as little as 512k of memory or as much as eight megs –with many intermediate configurations possible (particularly the six megabyte configuration, most desirable for use with a BridgeCard™). **8-UP!** is speedy, efficient, custom memory technology for your Amiga 2000 –and it's available now! **8-UP!** suggested list price is $199 (0k installed). Optional **PopSIMM's** are $49.95 per pair.

The 8-UP! photo shows the card half populated with conventional SIMM modules and half with MicroBotics PopSIMM's. PopSIMM's (without DRAM installed) are available as separate purchase items.

MicroBotics, Inc.

Great Products Since the Amiga Was Born!

811 Alpha Drive, Suite 335, Richardson, Texas 75081 (214) 437-5330

SOLD ONLY THROUGH YOUR AMIGA DEALER

Tell your dealer he can quick-order from MicroBotics directly – no minimum quantity –show him this ad!

4. Search the buffer forward (F6) or backward (F5) for a match on a partial string.

5. If you've entered a long line and accidentally hit the up or down cursor keys (thereby recalling a previous line), you can get it back with Control - (minus).

C. **Window Manipulation**

1. Shrink (F1) or expand (F2) the console window.

2. Pop the window (F10) or the screen (F9) to the back or to the front.

3. New window attribute options for including or excluding the gadgets and other features. (Very Nice!)

4. Clear the window with Control W.

Included with all of this is several utility programs for manipulating the command buffer in various ways. THIS is the type of console the Amiga should have had to begin with. Once installed, ConMan is transparent to the user. Each NEWCLI command will automatically engage the mastery of ConMan when the window is opened. I commend Mr. Hawes for his fine effort to bring this most welcome addition to the Amiga. He has also written a program he calls WShell which I will be using soon. I will cover WShell, Matt Dillon's Shell and, if available by then, AmigaShell (WB 1.3) in a future issue. A Shell is a command environment with some special features to make CLI operations much easier. This should not be confused with console handling software such as ConMan which gives you a better ANSI Terminal.

All of the escape sequences described in the last issue are still accessible and I've found ConMan to be compatible with everything I use. The latest version of ConMan may be found on Fred Fish Disk #133 or you can order it directly from the author -

William S. Hawes
P.O. Box 308
Maynard, MA 01754

ConMan is distributed as Shareware and the author requests a donation of $10.

I stumbled across an interesting window manipulation feature using the F1 and F2 keys as described above. If you create a window with NEWCLI that is some portion smaller than the full screen, the F2 key may be used to increase the window size to full screen while the F1 key will collapse the window. Try this:

```
1> NEWCLI CON:0/200/640/200/Neato
```

Now press F2. The window opens to full screen. Press F1 and it collapses into a small window at the top left corner. Press F1 to open it to full screen again. Press F2 and the window returns to the original size when opened. Press F1 and it will collapse and stay in the middle of the screen. What's so great about this? If you want a smaller window but need a full screen display once in a while, you don't need to have several console windows hanging around. Mr. Hawes indicates that the next release will include "an improved iconify state" which I hope will work similar to the next program described.

wIconify

Wouldn't it be nice if you could click on a gadget or something and cause a console window to get out of the way? We're all used to Icons. Why not provide a way to make a window disappear and leave an Icon behind to let you pop it back to full size when needed? Davide P. Cervone apparently felt that this would be a nice feature and wrote wIconify to solve the problem. Great job Davide! Several attempts have been made to get the window out of the way, from TinyWindows to SmartIcon. I feel wIconify provides the most elegant solution. Once installed, nearly any window on the Workbench screen will Iconify, even some commercial program windows. Because the program operates on the Workbench screen, you must run LoadWB before wIconify is started. To collapse a window press and hold the left mouse button then tap the right mouse button.

Although this works well enough, I think it might be better to provide a new window gadget instead. When collapsed, a small icon resembling a console window appears near the bottom of the Workbench screen with the title of the window underneath it. You can manipulate it the same way you would a Disk icon. New menu items are added to the Workbench menu for manipulating the icons. Below the DISCARD item on the WORKBENCH pop-down menu you should find an ICONIFY item which produces VERSION, CLEANUP, OPEN ALL, ORGANIZE, and, END. CLEANUP and ORGANIZE will neatly arrange multiple icons similar to the Clean Up selection on the SPECIAL pop-down menu. OPEN ALL will cause all iconified windows to open back up. END terminates wIconify and VERSION displays information about the program. You may also use the OPEN and CLOSE items from the WORKBENCH pop-down menu as you would with other Workbench icons.

Although Davide warns that he cheats some to make the program work and that it may not be compatible with a newer release of Workbench, my spies tell me that only the menu items seem to be inoperative with AmigaDOS 1.3.

MOUNTED

One of the problems we ran into while writing batch files was in determining what volumes are mounted without causing that blasted requestor to pop up. Bryce Nesbitt gives us a program called MOUNTED that eliminates the problem. MOUNTED will test for the presence of the volume specified in the command string and return a value that may be used by the IF command. The format is:

```
MOUNTED <volume:>
IF NOT WARN
   (command to execute if the volume is present)
ENDIF
```

(continued)

The Best.

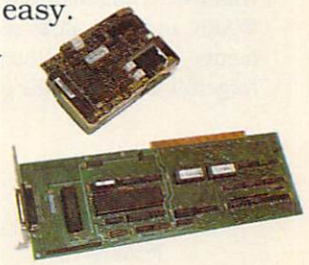
No Matter How You Stack 'Em.

No matter how you stack 'em, SupraDrives are the best. The best performers. The best designs. The best fit for your system.

And frankly, the best choice you could make, no matter how you stack your options.

Only Supra offers you complete SupraDrive Hard Disk Systems for the Amiga 500, the Amiga 1000, *and* the Amiga 2000 in various capacities and forms. You can get external drives, internal drives, interface kits, and RAM expansion kits. You can get drives ranging in size from 20MB all the way up to 250MB. Whatever your system configuration and needs, you can get a Supra hard disk that's just right for YOU.

And if what you need is backup or unlimited storage, you can get the SupraDrive FD-10, a new concept in mass storage that combines the best features of hard disks and floppy disks. The SupraDrive FD-10 lets you store a full 10MB of data on one floppy disk and then easily take the disk with you anywhere. Your storage capacity is limited only by the number of disks you have, and backups are quick and easy.



SupraDrive Interface for the Amiga 2000

When you buy your new disk drive from Supra, you'll be buying from a company that has years of experience to share with you and the stability to ensure that they'll be around when you need it. Only Supra Corporation offers you state-of-the-art, innovative technology plus experience and service you can count on.

Call your local dealer or:

Supra Corporation

1-800-727-8772 (Orders Only) or 503-967-9075
1133 Commercial Way, Albany, OR 97321
Telex: 5106005236 (Supra Corp.)
FAX: 503-926-9370

This could be expanded to allow you customize the environment easily. Here is an example script for starting a paint program.

```
MOUNTED Paint:
IF NOT WARN
   MOUNTED Pictures:
   IF NOT WARN
            ASSIGN HR: Pictures:HiRes
            ASSIGN LR: Pictures:LoRes
            ASSIGN BRUSH: Pictures:Brushes
   ENDIF
   RUN Painter
ENDIF
```

If the volume Paint: is not present, nothing happens. No requestor pops up and the batch file continues on. If it is present, then we test for the data disk. If the data disk is present then make some assignments and finish up by running the paint program. You can see another problem right away. What if you forgot to put the data disk in the drive? No assignments are made but the paint program is still started. How can we improve this? Read on. You will find MOUNTED on Fred Fish Disk #79.

ARP

Another problem that arises when working with batch files is that there is no easy way to make decisions based on user input. This would be handy in the script just shown. Although there are several P.D. programs that address this problem, I prefer the flexibility provided by the ASK command found in the ARP collection. I will be covering the AmigaDOS Replacement Project commands next issue. ASK allows you to suspend execution and wait for input from the keyboard. The syntax is:

```
ASK Prompt WARN OK TIMEOUT
```

Prompt is a string that will be displayed in the console window. TIMEOUT is a set time limit in seconds. To test the WARN and OK values we can use IF WARN and ELSE statements. Let's modify the script shown above to handle a forgetful user. For this purpose the file is saved as "Paint.stu".

```
MOUNTED Paint:
IF NOT WARN
    MOUNTED Pictures:
    IF WARN
        ECHO "The Data disk is not mounted."
     ECHO "Place volume Pictures: in any drive and enter"
     ASK "C to continue or A to abort." "C" "A" TIMEOUT 10
        IF WARN
            EXECUTE Paint.stu  ; execute this file again
        ELSE
            ECHO "ABORTED!"
            SKIP Exit
        ENDIF
    ELSE
        ASSIGN HR: Pictures:HiRes
        ASSIGN LR: Pictures:LoRes
        ASSIGN BRUSH: Pictures:Brushes
    ENDIF
    RUN Painter
ENDIF
LAB Exit
```

Now, if the the data disk is not mounted we ask the user what to do. If the WARN string is entered (C) the WARN condition is set to true and the file is executed again from the top. (This makes sure that the volume WAS mounted by testing again.) If the OK string is entered (A), the TIMEOUT value expires, or the RETURN key is pressed, the ELSE section is executed and it skips to the bottom. If something other than the WARN or OK strings is entered, ASK will re-display the prompt and wait for a valid response. The default values for Prompt, WARN, and OK are "?", "Y", and "N" respectively so it's not mandatory that you specify a string for each. Most of the other available ASK type of programs will only work with a "Y" or "N" answer. The ability to define a specific keyword to match for the True and False answers is what sets this ARP command apart from the rest.

ASK and the rest of the ARP commands can be found on Fred Fish Disk #123 or may be ordered from:

Microsmiths, Inc.
P.O. Box 561
Cambridge, MA 02140

DefDisk

I have shown you a method for changing the Assignments needed by AmigaDOS to make another disk your system (Workbench) disk. A quicker way to do this is to use the DefDisk command. By providing the name of the volume to change the assignments to, DefDisk will re-assign the logical names needed by Workbench to point to the new volume or directory.

```
DefDisk <volume name>
   e.g.
DefDisk DH0:
```

This is a lot easier and just as flexible as the original method. DefDisk may be found on AMICUS Disk #18.

I'll be covering batch files in detail in an upcoming issue. With the help of these and other Public Domain commands we will discover some of the power that the Amiga provides. Next issue I will discuss the differences and improvements in command line processing provided by the ARP command collection.

If you have any questions about the programs I discussed in this column or about other programs that pertain to the CLI that you would like to see covered, let us know.

Send your questions to:

Rich Falconburg
c/o *Amazing Computing*
P.O. Box 869
Fall River, MA 02722

•AC•

AC/BASIC 1.3

A compiler that makes AmigaBASIC a viable development language!

by Bryan Catley

Release 1.3 of Absoft's AC/BASIC compiler for the Amiga is now shipping. Registered users should receive the upgrade automatically. This release contains no external differences, but instead concentrates on improving compatibility with AmigaBASIC and fixing bugs. Additionally, a number of routines were rewritten to increase execution speed. The result is a compiler that makes AmigaBASIC a viable development language!

The distribution disk includes a number of examples, plus a full set from the reference manual. Also included is a short AmigaBASIC program that creates a HAM screen and then fills the screen with pixels drawn in random colors. There are two problems with this program you may wish to correct before using it.

First, change both "CALL CleanUpEverything()" statements to "CALL CleanUpEverything", and change the "SUB CleanUpEverything() STATIC" to "SUB CleanUpEverything STATIC". If you do not make this change, and you run the program under the interpreter, you receive a syntax error.

Second, change the two "pen% = INT(RND*31+1)" statements to "pen% = INT(RND*63+1)". This change allows the full effect of HAM colors to be seen. As originally written, only the blue intensity ever gets modified (and even then, only when the random number is above 15), so the result is a screen of mostly standard colors! The difference is dramatic!

Other example programs are the BSpread AmigaBASIC spreadsheet (AC V3.1.) and programs that show how to access command line or Workbench arguments from a compiled program. AC/BASIC also supports a number of language extensions, including the CASE statement. If you have been using this extension, you should be aware of some usage rule changes—made at the request of Microsoft Corporation—with this release.

My original review of AC/BASIC Release 1.2 (AC V2.9, V2.10.) included a list of all the bugs I encountered. With one exception, all those bugs have been fixed. The exception is that collision detection is still not handled correctly. As indicated, many other changes have been made for increased compatibility, and the mouse handling routines are among those which were completely rewritten. They now work beautifully... and all the time!

The documentation has also been changed; rather than a small three-ringed binder, it is now a standard soft bound book. Its contents have not been changed much, but some additional examples have been added. The print is also larger and easier to read. Overall, the documentation is superb, and you will probably find yourself using it rather than the AmigaBASIC manual (even when using the interpreter).

If you are not familiar with the compiler, Absoft does not charge royalties for distribution of compiled programs. They do request that you mail in the licensing agreement included in the package before distributing a program.

This release of AC/BASIC is really worth having! If you do any AmigaBASIC programming, buy it. (It's not expensive!)

AC/BASIC Compiler V1.3
Retail price $50
Absoft Corporation
2781 Bond Street
Auburn Hills, MI 48057
(313) 853-0095 or (313) 853-0050

•AC•

Extending AmigaBASIC

Accessing Intuition Functions From AmigaBASIC

by John Kennan

For about 2 years, I have been an active member in the Tri-City Amiga User's Group in Midland, MI. As the current president, part of my role is helping group members use their Amigas more effectively. Recently one of our members approached me and asked if it were possible to create borderless windows in AmigaBASIC. He was working on a project involving overlaying text on a video signal using a genlock and an Amiga. The genlock substituted the video signal for the background color in Amiga windows and screens. An AmigaBASIC program generated text for the video. The problem was that windows created from within an AmigaBASIC program always included a window border which was visible in the final videotape. I took a quick look through the AmigaBASIC manual and found that AmigaBASIC gave the user very few options on window management. None of the available options allowed creating borderless windows. Since AmigaBASIC always creates windows with borders, I needed to come up with a way of making a window border transparent. The simplest way would be to tell the Amiga to make the border color the same as the background color. Once again, AmigaBASIC offered no obvious way of doing this.

After studying the problem, I realized the only way to remove a window border was to access Intuition data structures and functions. Unfortunately, this advanced topic is not covered in the AmigaBASIC manual. This article attempts to remedy this problem by demonstrating the procedure for calling some of the Intuition routines. Intuition.bas, the program accompanying this article, includes some of the most useful calls. The program name is derived from the fact that the routines are available as Rom Kernel calls when the Intuition Library is opened.

For those of you in a hurry, I'll start with a quick description of how to get Intuition.bas up and running, and then we'll go into a brief discussion on how the program works.

Before typing in listing 1, you'll have to do a few things. For AmigaBASIC to access Intuition calls, the Libs: directory of your Workbench must contain a file named 'intuition.bmap' . Unfortunately, it is very likely that you don't have that file. Don't worry, all is not lost. The file is very easy to create using a BASIC program found in the BASICDemos drawer of the Extras disk that came with Workbench 1.2. Simply load the program ConvertFD and enter 'run'. The program will prompt you for the name of an .fd file to convert. At this point, enter 'Extras:FD1.2/intuition_lib.fd'. The program prompts you for the name of a .bmap file to create. If you have a dual drive system, you can have ConvertFD create the file directly in the Libs: directory by entering 'Libs:intuition.bmap'. If you have a single disk drive, you can save yourself a lot of disk swaps by entering 'Ram:intuition.bmap'. The file intuition.bmap will be created in the Ram: disk. It is then a simple matter of going to a CLI window and entering 'Copy Ram:intuition.bmap to Libs:'.

If you already have a copy of the file intuition.bmap, you may want to go through the trouble of creating a new version of the file. The reason is that the file 'intuition.bmap' contains information that tells AmigaBASIC how to call Intuition functions by name. Older file versions might not include function calls added to version 1.2 (or 1.3) of the operating system.

Now type in and save listing 1. Intuition.bas is a program to demonstrate some of this new function calls. Included are routines for making window borders invisible, resizing windows, moving windows, and moving screens. To see the demonstration, enter 'run' .

Understanding how this program works will allow you to better understand the inner workings of the Amiga. This should allow an AmigaBASIC programmer to explore other Library calls. Furthermore, if at some future time you decide to make the transition to another programming language such as C, Forth, Modula-2, or assembly language, this information will help you get started.

Most of the Amiga operating system's inner workings revolve around structures. In the "C" programming language, a structure type is first defined as having a certain format. The format defines the number of bytes of memory an occurrence of the structure requires, and the sequence and size of the data to be stored in the structure. After that the program can allot space for an occurrence of the structure and store data in the structure according to the structure's format. Any program can then access the information if it knows the format and the starting address of the structure.

A powerful feature of structures is that they will often contain addresses which point to other structures (thus a 4 byte address is referred to as a pointer). In this way structures can be linked together through a series of pointers. Thus a single structure can form the beginning a linked group of structures (sometimes referred to as a linked list). Linked lists of this type are fundamentally import to the Amiga Rom Kernel in keeping track of what occurs in the Amiga multi-tasking environment.

An example of one of the most important structures available in Intuition - the Window structure will make this clear. The Amiga uses this structure to keep track of information pertinent to window management. Rather than describe the entire structure (it contains 48 different entries), we'll just look at some of the highlights to get a flavor for the type of informa-

tion stored in this particular structure. Think of the Window structure as an address in memory at which the following information is stored:

bytes	'C reference'	explanation
0-3	NextWindow	the address of another window structure
4-5	LeftEdge	the left edge of the window
6-7	TopEdge	the top edge of the window
8-9	Width	the width of the window
10-11	Height	the height of the window
12-13	MouseY	mouse position
14-15	MouseX	mouse position
16-17	MinWidth	the minimum width of a resizable window
18-19	MinHeight	the minimum height of a resizable window
20-21	MaxWidth	the maximum width of a resizable window
22-23	MaxHeight	the maximum height of a resizable window
24-27	Flags	Intuition flags describing attributes of the window
28-31	MenuStrip	A pointer to a structure describing the menus for the window
32-35	Title	A pointer to the title of the window
36-39	FirstRequest	A pointer to a requester structure
40-43	DMRequest	A pointer to a double click requester structure
44-45	ReqCount	Number of requesters currently active which are blocking window input.
46-49	Screen	A pointer to a structure describing the screen in which this window appears. This value is used by ScreenToFront and ScreenToBack.
50-53	RPort	A pointer to a rastport structure which describes the contents of the window. This is the number returned by the BASIC command Window(8). The number is used by many of the Rom Kernel graphics library commands.
.		
.		
.		
98	DetailPen	A number representing the color register used to draw gadgets or text in the title bar.
99	BlockPen	The color register use for area fills and line rendering in the window border.

Notice that the above structure contains a variety of information. Data within a structure may be stored as a byte, short word (two bytes), long word (4 bytes), or just about anything else. Fortunately AmigaBASIC includes the commands POKE, POKEW, POKEL, PEEK, PEEKW, and PEEKL which allow a programmer to easily access data in a structure no matter how the data is stored.

The most important thing to remember about structures is that while there are many structure types (i.e. Window, Screen, Gadget, Requester, etc. are all structures defined by Intuition), once a structure type is defined, all structures of that type will have the same layout. Thus there may be many Window structures resident in memory, but the 10th and 11th byte of each of those structures will always describe the height of that Window structure's window.

Anyway, a structure's real power is in passing information from one part of a program to another part, such as a subroutine. Rather than passing the 48 variables defined in the Window structure, we can pass just one number—the starting address of the Window structure. The subroutine can then extract any information it needs from the structure simply by knowing the offsets that define where the information is stored. Most of the Intuition Rom Kernel routines require as input the address of the Window structure that describes the window of interest.

In principle, all of the Intuition Library calls could be implemented from AmigaBASIC. In practice, most of the calls require a great deal of preparation which might make them difficult to use. We will focus on the ones which are easy to implement and offer the added benefit of overcoming some of AmigaBASIC's more serious limitations. For example, once you create a window or screen from within an AmigaBASIC program, the program has virtually no control over that window or screen. Fortunately, the Intuition library calls change this situation dramatically. Below is a list of the calls used in listing 1. Within the program, Rom Kernel routines are called by name. The more useful calls (from a BASIC programmer's point of view) are as follows:

WindowToFront(MyWindow&) - Moves the specified window to the front of the screen

WindowToBack(MyWindow&) - Moves the specified window to the back of the screen.

RefreshWindowFrame(MyWindow&) - When you move or resize a window under program control, Intuition sometimes gets confused and fails to redraw the window's border. This routine was added to version 1.2 of the operating system to allow programmers to force Intuition to redraw the border. This routine is useful as it allows a programmer to modify the

(continued)

Window structure and then force the Amiga to redraw the border. This is how we make the border disappear in the demo. ScreenToFront(MyScreen&) - Moves the specified screen to the front of the display.

ScreenToBack(MyScreen&) - Moves the specified screen to the back of the display.

MoveWindow&(MyWindow&,dx%,dy%) - Moves the window by the specified number of pixels left or right and up or down.

SizeWindow&(MyWindow&,dx%,dy%) - Increases or decreases the size of the window by the specified number of pixels left or right and up or down.

In the calls listed above, function's name is followed by its required parameters. I have substituted variable names for the parameters. The '&' symbol indicates either that the variable is a long integer, or that the ROM call returns a long integer. The '%' symbol indicates that the variable is a short integer.

All of the calls listed above require the address of the window's window structure as a parameter (as indicated by the variable MyWindow&). This address is readily obtained by forcing the window to become the current window with the AmigaBASIC command WINDOW OUTPUT <window-id>, and then getting the address of the window structure with the statement WINDOW(7). Save this value in a variable for later use. Listing 1 includes examples using all the above calls.

All the calls become available when the Intuition Library is opened with the LIBRARY statement. AmigaBASIC restricts use of Library calls. If a Library call has multiple parameters or if it returns a value, it must be defined in advance with a DECLARE FUNCTION statement. Furthermore, once a call is defined in a DECLARE FUNCTION statement, it can only be used in an arithmetic expression. For example, the following program will open the intuition library and then decrease the size of the default AmigaBASIC window. (Make sure the window is open and of sufficient size so that shrinking the window 5 pixels in the x and y directions won't crash the machine).

```
LIBRARY "intuition.library"
DECLARE FUNCTION SizeWindow& LIBRARY
WINDOW OUTPUT 1
MyWindow&=WINDOW(7)
x&=SizeWindow&(MyWindow&,-5,-5)
LIBRARY CLOSE
```

Exercise caution when using Intuition ROM Kernel routines. Remember ROM calls are outside of the control of AmigaBASIC and errors will not result in the relatively benign AmigaBASIC error codes. To recover from an error in a ROM Kernel call, you will probably have to reboot the machine. To avoid errors, make sure the window to be operated on is open and that the parameters passed to the routine are in an acceptable range.

Listing one also includes some subprograms which make ROM Kernel calls a little easier to use. CALL SetWindowSize(MyWindow&, WWidth%, WHeight%) will set the window to the desired width and height. The advantage of SetWindowSize over SizeWindow is that SetWindowSize accepts the desired window size as parameters and then does the required error checking before calling the ROM Kernel routine SizeWindow. CALL KillBorder(MyWindow&) and CALL RestoreBorder(MyWindow&) are used for removing or reinstating the border on an AmigaBASIC window.

By the way, KillBorder and RestoreBorder only effect the border. These subprograms will not effect gadgets within the window border. Therefore if you want the border to completely disappear, create windows without gadgets (read over the Window command in the AmigaBASIC manual). It is also advisable to leave out the Window Close gadget since this prevents the user from closing a Window the program may later try to manipulate.

Listing two contains a small demo program which I used at our club meeting to demonstrate borderless windows in AmigaBASIC. The program superimposed scrolling text on a background video signal using a genlock device. As an aside I might mention that in my first version of this program I used the BASIC SCROLL command. I discovered that repeated use of the SCROLL command causes the Amiga to use up memory rapidly. Furthermore, the memory is not deallocated when the BASIC program is terminated. Left on its own, a program repeatedly using the SCROLL command will crash the machine. I suspect that the problem originates in a bug in the Rom Kernel routine ScrollRaster. Hopefully this will be fixed in the next revision of the operating system.

That about covers our introduction to using LIBRARY calls from AmigaBASIC. If you want to learn more about the ROM calls that are available, you might want to examine the FD files in the FD1.2 drawer on the Extras disk. All Rom Kernel routines are listed in these files. Learning how to use them might be a little tricky. You might want to seek help from other

(continued)

zo·ë·trope (zō′ə trōp)

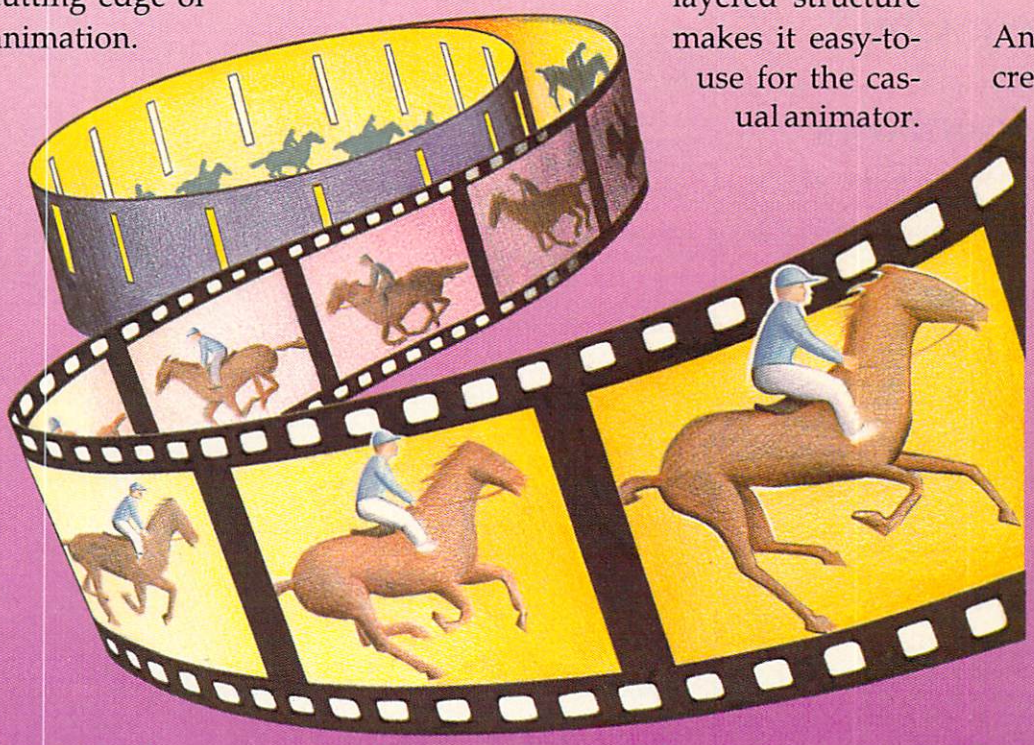
n. [irreg. f. Gr. *zoe-*, life + *tropos*, turning] 1. a device that gave static images an illusion of motion. Known as the "wheel of life", the zoetrope brought the magic of animation to the parlours of the mid-1800's. **2. a new state-of-the-art advanced animation system for the Amiga® computer from Antic Software.**

Now! The program you've been hearing so much about... Zoetrope puts you right on the cutting edge of animation.

Its high end features satisfy even the most demanding of professional users. Its unique layered structure makes it easy-to-use for the casual animator.

Bring your imagination to life with Zoetrope: the faster way to animate.

Now with Zoetrope, The Animation System™, *you* can create magical motion.



Here are just a few of Zoetrope's exciting, advanced features:

- Spin and swoop titles and images through 3D space.
- Apply computer power to classic "cel" animation.
- Overlay and underlay (compositing) your favorite pictures or animations.

Taking animation into the 21st Century

By Jim Kent, the creator of Aegis Animator*

ZOETROPE
THE ANIMATION SYSTEM™

Another great Cyber Graphics product.

To see the entire line of Cyber Graphics from Antic Software, visit your favorite Amiga® retailer.
To order direct, call 800-234-7001.

MM8AW3

Antic Software

Listing One

```
REM Intuition.Bas - John Kennan
REM A program to demonstrate the various functions
REM available throught the  use of Intuition Library Calls
REM Some of the Intuition functions we can use
REM RefreshWindowFrame, WindowToFront, WindowToBack,SizeWindow
REM ScreenToBack, ScreenToFront, and MoveWindow

REM DECLARE FUNCTION should only be necessary if the Library
call
REM returns a value.  This does not seem to be the case.
REM If the intuition library call returns a value
REM or if it requires more than one parameter
REM we have to declare it as a function

DECLARE FUNCTION SizeWindow& LIBRARY
DECLARE FUNCTION MoveWindow& LIBRARY

REM The next line requires that the file
REM 'Intuition.bmap' be present in
REM the Libs: directory of your workbench disk.

LIBRARY "intuition.library"
WINDOW OUTPUT 1

REM First we get the addresses of the window and screen
REM structures for the default AmigaBasic window

MyWindow1&=WINDOW(7)
MyScreen0&=PEEKL(MyWindow1& + 46)

REM Now we open a new screen and 2 new windows
SCREEN 1,320,200,2,1
WINDOW 2,"Window2",(10,10)-(250,150),16,1
WINDOW 3,"Window3",(15,15)-(280,180),16,1

REM Next get the addresses of the window and screen
REM structures for the new windows and new screen
WINDOW OUTPUT 2
MyWindow2&=WINDOW(7)
WINDOW OUTPUT 3
MyWindow3&=WINDOW(7)

MyScreen1&=PEEKL(MyWindow3& + 46)

REM Now lets try out some new commands
PRINT "This is a standard"
PRINT "AmigaBasic window"
PRINT "in a standard AmigaBasic"
PRINT "screen"
SecsToWait=1: GOSUB TimedWait

PRINT "Let's move this window"
PRINT "Behind the other window"

SecsToWait=1: GOSUB TimedWait
WindowToBack(MyWindow3&)
SecsToWait=1: GOSUB TimedWait

WINDOW OUTPUT 2
PRINT "And we can go back"
PRINT "to the other window"
SecsToWait=1: GOSUB TimedWait

WINDOW OUTPUT 3
WindowToFront(MyWindow3&)
SecsToWait=1: GOSUB TimedWait
PRINT "I'm back"
SecsToWait=2: GOSUB TimedWait

PRINT "Let's get rid of this"
PRINT "annoying border"

SecsToWait=2: GOSUB TimedWait
CALL KillBorder(MyWindow3&)
SecsToWait=1: GOSUB TimedWait

PRINT "And now let's get the border back"
SecsToWait=2: GOSUB TimedWait
CALL RestoreBorder(MyWindow3&)
SecsToWait=1: GOSUB TimedWait

CLS
PRINT "Now let's make the window"
PRINT "smaller"

SecsToWait=2: GOSUB TimedWait
CALL SetWindowSize(MyWindow3&,240,60)
SecsToWait=1: GOSUB TimedWait

PRINT "We can also make the window"
PRINT "big again"

SecsToWait=2: GOSUB TimedWait
CALL SetWindowSize(MyWindow3&,240,140)
SecsToWait=2: GOSUB TimedWait

PRINT "We can even make the"
PRINT "Window Move"

SecsToWait=1: GOSUB TimedWait
x&=MoveWindow&(MyWindow3&,20,20)
SecsToWait=1: GOSUB TimedWait
x&=MoveWindow&(MyWindow3&,-20,-20)
SecsToWait=2: GOSUB TimedWait

PRINT "How about checking on"
PRINT "the other Screen"

SecsToWait=2: GOSUB TimedWait
ScreenToFront(MyScreen0&)
SecsToWait=1: GOSUB TimedWait

WINDOW OUTPUT 1
PRINT "Not much going on here"
PRINT "Let's go back and finish up"
SecsToWait=2: GOSUB TimedWait
```

```
ScreenToFront(MyScreen1&)
SecsToWait=1: GOSUB TimedWait
WINDOW OUTPUT 3
PRINT "Time to close up shop"
SecsToWait=1: GOSUB TimedWait

WINDOW CLOSE 3
WINDOW CLOSE 2
SCREEN CLOSE 1

WINDOW OUTPUT 1
PRINT "All done"

LIBRARY CLOSE

END

SUB KillBorder(MyWindow&) STATIC
   blockpen&=MyWindow& + 99
   POKE blockpen&,0
   RefreshWindowFrame(MyWindow&)
END SUB

SUB RestoreBorder(MyWindow&) STATIC
  blockpen&=MyWindow& + 99
  POKE blockpen&,1
  RefreshWindowFrame(MyWindow&)
END SUB

SUB SetWindowSize(MyWindow&,WWidth%,WHeight%) STATIC
  CurWidth%= PEEKW(MyWindow&+8)
  CurHeight%=PEEKW(MyWindow&+10)
  MinWidth%= PEEKW(MyWindow&+16)
  MinHeight%=PEEKW(MyWindow&+18)
   MaxWidth%= PEEKW(MyWindow&+20)
  MaxHeight%=PEEKW(MyWindow&+22)
  IF ((WWidth% < MaxWidth%) AND (WWidth% > MinWidth%)) THEN
    DeltaWidth%=WWidth% - CurWidth%
    ELSE
     DeltaWidth%=0
  END IF
  IF ((WHeight% < MaxHeight%) AND (WHeight% > MinHeight%)) THEN
    DeltaHeight%=WHeight% - CurHeight%
    ELSE
    DeltaHeight%=0
  END IF
 x&=SizeWindow&(MyWindow&,DeltaWidth%,DeltaHeight%)
END SUB

REM I tried to make the next routine a subprogram,
REM but ON TIMER(n) GOSUB
REM doesn't appear to work from within a subprogram

TimedWait:
  IF SecsToWait<1 THEN RETURN
  ON TIMER(SecsToWait) GOSUB EndTimer
  TIMER ON
  REM The next line is necessary because other events besides
the
  REM timer will cause AmigaBasic to continue
  WHILE SecsToWait<>0:SLEEP:WEND
RETURN

EndTimer:
   SecsToWait=0
   REM Now we disable the timer event
   TIMER OFF
RETURN

REM Display.Bas - John Kennan
REM Displays scrolling text ON a borderless WINDOW

LIBRARY "intuition.library"

SCREEN 1,640,200,2,2
WINDOW 2,"Window2",(0,0)-(631,185),16,1
ON BREAK GOSUB ExitProg
BREAK ON

WINDOW OUTPUT 2
MyWindow2&=WINDOW(7)

MyScreen1&=PEEKL(MyWindow2& + 46)
CALL KillBorder(MyWindow2&)

ArraySize=6+((144-136+1)*2)*INT((600-15+16)/16)*2
DIM Array%(ArraySize)
text$="  Welcome to the TRI CITY AMIGA User's Group  "
counter=2
WHILE INKEY$=""
  LOCATE 18,72
  PRINT MID$(text$,counter,1)
  counter=counter+1
  IF counter>LEN(text$)-1 THEN counter=2
  FOR n=1 TO 4
    GET(17,136)-(600,144),Array%
    PUT(15,136),Array%,PSET
  NEXT n
WEND

ExitProg:
   WINDOW CLOSE 2
   SCREEN CLOSE 1
   LIBRARY CLOSE
END

SUB KillBorder(MyWindow&) STATIC
   blockpen&=MyWindow& + 99
   POKE blockpen&,0
   RefreshWindowFrame(MyWindow&)
END SUB

SUB RestoreBorder(MyWindow&) STATIC
  blockpen&=MyWindow& + 99
  POKE blockpen&,1
  RefreshWindowFrame(MyWindow&)
END SUB
```


•AC•

Converting Patch Librarian Files

(or, "How to get your sounds from there to here")

by Phil Saunders

When I bought my Yamaha DX7 synthesizer, I also bought a used patch editor for the Macintosh. The musician I bought it from told me he had, "about a thousand patches", stored on the disk. I started using the DX7 with my Amiga 1000 and a MIDI interface, but the thought of those thousand patches haunted me. I knew there must be a way to use them with my Amiga.

One afternoon I hooked my girlfriend's Macintosh to my Amiga with a null modem cable and transferred all the patch files. Now I had the files on an Amiga disk, but my Amiga patch editor refused to recognize them. Whenever I tried to load the Macintosh files, my patch editor showed a screen full of garbage. I soon realized that the Macintosh editor stored DX7 patch data in a different format from Dr. T's DX-Heaven. What I needed was a way to convert the Macintosh files into Dr. T's files. After some research, I was successful.

The techniques I used can be modified to translate sounds for any synthesizer to Amiga format, regardless of the editor or computer used to save the patches. The same techniques can also be used to convert other kinds of data for use in Amiga programs. Here's how to go about the task.

Compare 'em

The first step is to compare a few of your existing Amiga patches to the "foreign" patches. In my case, after LISTing both files, I saw that the Macintosh file was 4352 bytes long, while the Amiga file was 8192 bytes long. Checking Howard Massey's, "The Complete DX7", I discovered that the Yamaha format for a bulk voice dump is 4096 bytes long (128 bytes times 32 voices). "The Complete DX7" also showed how the various parameters of each voice were stored within those 4096 bytes. I now knew the Yamaha format for a voice dump. (This information is usually in the back of a synthesizer manual, under the MIDI implementation section or the system exclusive code section).

The next step is to compare both patch files to the synthesizer's own patch format. You could use a disk or file editor to do this; I used the TYPE command with the OPT H extension. (TYPE AmigaFile to PRT: OPT H). This gave me a printout of both the hexadecimal and the ASCII codes of each file. (See Listings 1 and 2.) The TYPE command is useful because it provides two ways of looking at every byte in the file. The ASCII printout on the right side allows you to scan the file for intelligible information; the hexadecimal values on the left let you get precise values of the data.

In this case we can quickly locate the name of the first patch in the Amiga file. Yamaha DX7 patch names have ten characters in ASCII format. Looking at the right column we see that "AC.PIANO" starts at $76 and ends at $7F. The ASCII characters in the patch name are a good clue to how the patch data is stored. Examination of the Amiga file shows that voice names recur every 128 bytes until location $0FFF. (The first name is at $76-$7F, the second name is at $F6-$FF, the third is at $176-$17F, and so on through the thirty-second, which is at $FF6-$FFF).

Since a DX7 patch takes up 128 bytes in bulk dump format, the patch data is clearly stored in this first part of the file. If we compare our listing of the Amiga file to the DX7 system exclusive data in the book, we see that DX-Heaven appears to store data in the same order as the DX7 system exclusive commands. (Most voice editors will conform to the manufacturer's bulk dump format because they need to send the data to the synthesizer in that format.) Looking at the ASCII dump after $1000, we see text strings showing the performance function data for the DX7. Since this information is not included in the 32 voice bulk data dump, we will ignore it. (Some synthesizers do save performance data with their patches. Usually this data will be part of the patch, so no special steps will be necessary to retrieve it).

What's your name?

We now know that the first 4096 bytes of the Amiga file are the same as the standard Yamaha format. But what about the Macintosh file? Let's try the same technique of looking for the patch names. The third through sixth bytes have some ASCII data, but they are surrounded by hexadecimal zeroes. That doesn't seem right, so we keep scanning. Looking further on, we see the string "A-2" starting at $F8-$101. The patch names in the Macintosh file also recur every 128 bytes, so we find the next name at $178-$181, followed by another at $1F8-$201, and so on. We'll assume that Opcode didn't devise a custom data format since the amount of space between patch names is the same as the Yamaha specifications. So, the Macintosh file looks like the Yamaha bulk dump format, but with a lot of extra data tacked on to the beginning.

In the Yamaha format, the patch name takes up the last 10 bytes of the 128 allocated for each voice. The first 118 bytes contain data for the other parameters in the voice. So, if we want to find the first byte of any patch, we need to look for the location of the first byte of its name and then subtract 118. If we do this for the Amiga file, we find the first byte of the patch name at $76 (decimal 118). If we subtract 118 from this value, we get zero. The file starts at $0000, so this makes sense.

The Macintosh file has its first patch name at $F8 (decimal 248). If we subtract 118 from 248, we get 130 ($82). This means that the first voice in the Macintosh file starts at $82 and runs for 128 bytes (to $101). The second goes from $102 to $181, with the last patch ending at $1081. We want to extract the data from $82 to $1081 (Decimal 130 to 4225) and discard the rest of the file. I wrote a simple utility, CONVERT, which will skip a user specified number of bytes into a file and then write a new file of the specified length.

Convert It!

To convert "Macfile" to Amiga format, we would use the following command: "CONVERT df0:Macfile df0:AmigaFormat.TX7 130 4096". ("Convert Macfile to AmigaFormat.TX7 by copying 4096 bytes after skipping 130 bytes".) The output file created by the CONVERT command is ready to be loaded into DX-Heaven or a public domain program like DX-fer. The actual conversion of the file is easy; the difficulty is in finding what bytes need to be skipped and which need to be kept. There are several standard steps in doing this kind of file conversion. I will summarize them in general terms.

1: Get hexadecimal dumps of the different file formats. The "TYPE OPT H" command is useful for this purpose. If you have access to the program that generated the file you wish to convert, try entering the same data in both programs and then comparing the resulting files. This is about the only way to do a conversion if a program rearranges data. (Few patch editors will do this).

2: Examine the "standard" format. Most manufacturers list their system exclusive data formats in the synthesizer manual. Even if the information is not listed in the standard manual, it may be available in other books or by request from the manufacturer. A public domain program that accepts raw MIDI system exclusive dumps is another way to get a look at what format the synthesizer expects.

3: Look for landmarks. In our example we looked for the ASCII codes that represented patch names. Once we found these we could look for patterns that recurred (patch names being 128 bytes apart, for example). This kind of analysis gives you useful information about how data is stored.

4: Compare the different files. Do the patterns repeat the same way in each file? Where is the "good" data stored? How do the files you are trying to convert differ from the standard format? These are the kinds of questions you should ask yourself.

5: Find the start and end points of the data you want. If you know that each voice in the file starts and ends a certain distance after your "landmark", you should be able to locate the data you want. The CONVERT utility is a very useful tool for extracting data—once you know which data you want.

6: Put the new file in proper format. This step is unnecessary in our example. But if we needed to add a "header" so that data could be read by an application program (a program that required IFF format samples, for example) we would have to create the header and then JOIN it to our new data file. One easy way would be to use CONVERT to extract an existing header, then use a disk editor to alter it to match the foreign file. The header could then be JOINed to the new file. You can also use CONVERT several times on a file to extract different data, then use JOIN to rearrange the pieces. This hasn't been necessary in my applications, but it is certainly possible. (See Richard Rae's article in the Sept. 1988 issue of Amazing Computing for more details on IFF format sounds).

A general rule for translating Yamaha DX7 files is to find the location of the first patch name and then subtract 118. The result is both the start of the patch data and the number of bytes that should be skipped. In the example cited in the article, the first patch name starts at $F8 (Decimal 248). If we subtract 118, we get 130 ($82). Since the first byte of the file is numbered at $0000, the byte located at 130 is actually the 131st byte in the file. By skipping the first 130 bytes and extracting the next 4096, we extract the patch data. I have compiled a list of skip values for several DX7 patch formats. Refer to Table 1 to get the correct value, then use the CONVERT command to convert the file to Amiga format. (Other synthesizers are handled in the same way, although the starting offset,118 in this case, will probably be different.)

The CONVERT utility gives you access to the thousands of public domain sounds that exist on bulletin boards across the country. I have used it to convert more than three thousand DX7 patches for use on my Amiga. It does require a little work, but once you learn how a file is stored, you can convert all files of that type with little additional effort. I hope you find CONVERT as useful as I do.

Listing One
Amiga DX7 File (Dr. T's Format)

```
0000: 5F1C1B2F 635A0000 05000004 3A003B02    _../cZ.....:.;.
0010: 0053001B 32635A00 00310000 00520C44    .S..2cZ..1...R.D
0020: 0200581C 1B32635A 00003100 00003B10    ..X..2cZ..1...;.
0030: 6202005F 1C1B2F63 5A000005 0063044A    B._../cZ....C.J
0040: 0C4F0A00 585C473F 63435B5A 27000D00    .O..X\G?cC[Z'...
0050: 4A045802 00581C1B 32635A00 00310000    J.X..X..2cZ..1..
0060: 003A1063 02000000 00003232 3232030D    .:.C......2222..
0070: 23000000 070C4143 2E504941 4E4F2020    #.....AC.PIANO
0080: 4B1D1604 635F0000 16521B00 34045306    K...C_...R..4.S.
0090: 004B1913 23635D00 0000000D 0051045A    .K..#C].....Q.Z
00A0: 02005018 1235635E 00000000 00003B08    ..P..5C^......;.
00B0: 63020058 18002B63 5B00001C 0033001B    C..X..+C[....3..
00C0: 044F0A00 40261123 635F0000 00000400    .O..@&.#C_......
00D0: 44045302 00511912 30636000 00000000    D.S..Q..0C`.....
00E0: 00330C63 02006363 63633232 3232020C    .3.C..cccc2222..
00F0: 23000000 01185049 414E4F20 39202020    #.....PIANO 9
(File continues...)
```

(continued)

Listing Two
Macintosh DX7 File (Opcode Format)

```
0000: 00046176 63310000 00000000 00000000     ..avc1..........
0010: 00000000 00000000 00000000 00000000     ................
0020: 00000000 00000000 00000000 00000000     ................
0030: 00000000 00000000 00000000 00000000     ................
0040: 004D5044 374D3250 38010000 00000000     .MPD7M2P8.......
0050: 00880000 00100200 0000009A 554C1D9A     ............UL..
0060: 554C1F00 00000000 00000000 00000000     UL..............
0070: 00000000 00000000 00008181 C7EE0000     ............C...
0080: 00203540 2C46635C 38000F19 0E083A08     . 5@,Fc\8.....:.
0090: 360E0F56 43264056 5C4A000F 3B000912     6..VC&@V\J..;...
00A0: 04540200 60131441 635C5900 00000005     .T..`..Ac\Y.....
00B0: 3A085804 19323423 39635C5B 003F623C     :.X..24#9C\[.?B<
00C0: 093A0463 02006347 2347525C 57003600     .:.C..cG#GR\W.6.
00D0: 00003900 5602323A 1E194163 5C5A0000     ..9.V.2:..Ac\Z..
00E0: 0000003B 04630000 63635F3C 32323232     ...;.C..cC_<2222
00F0: 0E0F0B00 03611918 412D3220 20202020     .....a..A-2
0100: 20206350 165A6363 63000000 00003800       cP.Zccc.....8.
0110: 60043263 50165A63 63630000 00000038     `.2cP.Zccc.....8
0120: 00600400 6350165A 63636300 00000000     .`..cP.Zccc.....
0130: 60006002 00635036 52636363 00000000     `.`..cP6Rccc....
0140: 00580060 04326314 165A6363 61000000     .X.`.2C..Zcca...
0150: 0A000800 63040163 50165A63 6363003C     ....C..cP.Zccc.<
0160: 63000028 00630000 4B504B3C 32323232     C..(.C..KPK<2222
0170: 1F082362 0000180C 12D4F52 47414E30     ..#B....A-ORGAN0
0180: 30365E00 63636300 00000000 00003800     06^.ccc.......8.
0190: 4207030A 02636363 00000000 00000038     B....ccc.......8
01A0: 0057052E 29006363 63000000 00000000     .W..).ccc.......
01B0: 38004500 15200063 63630000 00000000     8.E.. .ccc......
01C0: 0038003E 061C3200 63636300 00000000     .8.>..2.ccc.....
01D0: 00003800 45020907 161C4063 42000000     ..8.E.....@cB...
01E0: 00000038 00560600 00000063 47320063     ...8.V.....cG2.C
01F0: 110F2300 00000100 412E4643 45203330     ..#.....A.FCE 30
0200: 2E355825 100A635E 00632300 00003D1C     .5X%..C^.C#...=.
(File continues...)
```

Table One

Format	Skip Value	Length (before conversion)
Opcode	130	4352
MIDIEX	2	4224
Gen Patch	62	4170
DX Android	64	4161
Sonus	6	4104

Convert V1.0

CONVERT accepts the following command line:

CONVERT infile outfile skip (length)

infile is the file to be converted.
outfile is the file to be created.
SKIP is the number of bytes to be skipped.

(LENGTH) is an optional parameter giving the number of bytes to be transferred. If omitted, CONVERT will transfer the rest of the file. "CONVERT df0:infile df0:outfile 20 1000" would skip the first 20 bytes of df0:infile, then read the next 1000 and write them to df0:outfile.

```
/*    Convert is written by Phil Saunders
      with encouragement and advice from Don Curtis

      Convert expects the following command line:

      Convert infile outfile skip [length]

      infile is the file to be converted,
      outfile is the converted file,
      SKIP is the number of bytes to be skipped,
      LENGTH is the number of bytes to be copied
      If LENGTH is omitted, Convert will copy until it reaches the
      end of the input file.
*/

#include <stdio.H>

void main(argc, argv)
int argc;
char *argv[];

{
   FILE *infile, *outfile;

   int skipcount, length, data, ret;

      /* Are there 4 or 5 arguments from CLI ? */

      if (argc < 4 || argc > 5)  {  /*Check for illegal number of arguments*/
          printf("Bad Arguments...Usage: infile outfile skip [length]\n");
         exit(20);
      }
      else if (argc == 4)
          length = -1; /*no length parameter passed*/

      else  {
          length = atoi(argv[4]); /*get length parameter and convert to int*/

            if (length < 1)  {
                printf("Can't convert a negative number of bytes\n");
               exit(20);
            }
      }

       skipcount = atoi(argv[3]); /*get bytes skipped parameter and convert*/

          if (skipcount < 0)  {
              printf("Can't skip a negative number of bytes\n");
             exit(20);
          }

   infile = fopen(argv[1], "r");                /*open Input file*/
      if (infile == NULL)  {
          printf("Couldn't open input file\n");
          fclose(infile);
          exit(20);
      }

   outfile = fopen(argv[2], "wb+");             /*open Output file*/
      if (outfile == NULL)  {
          printf("Couldn't open output file\n");
          fclose(infile);
          fclose(outfile);
          exit(20);
      }

   ret = fseek(infile, (long) skipcount, 0);    /*Skip bytes in input file*/
      if (ret != 0)  {
          printf("Input file too short\n");
          fclose(infile);
          fclose(outfile);
          exit(20);
      }

   while (1==1) {
       data=fgetc(infile);  /*get input byte*/

       if (data == EOF)                /*check for EOF*/
          break;

       else
           ret = fputc(data, outfile);  /*write to output file*/
             if (ret == EOF)  {
                 printf("Error writing Output file\n");
                 fclose(infile);
                 fclose(outfile);
                 exit(20);
             }
       if (-length == 0)
          break;          /*Transferred length # of bytes, so exit*/
   }

      ret = fclose(infile);     /* close files*/
      ret = fclose(outfile);
}
```

•AC•

C Notes from the C Group

by Stephen Kemp, PLINK ID: SKEMP

Program or function control must often be determined by the value contained in a variable, or by the results of an expression. For instance, most programs that accept keyboard input have code that determines the program functions to execute—based on the keystroke. The evaluation of the keystroke determines what happens next in the program.

There are several ways to write the code to handle this type of program control. The first method that springs to mind is use of multiple "if" statements to check the conditions and then direct the program. This is a very logical choice, especially when there are only a few possible branches. But when there are many possible branches, the best method to control program flow may be the "switch" statement.

The Switch Statement

The "switch" statement evaluates an expression (or variable) and uses the results to match the expression with one of many branches. The syntax of the switch statement is:

```
switch (expression) {
   case constant 1: statements
   case constant 2: statements
   ...
case constant n: statements
   default:      statements
}
```

The expression determines branching within a switch statement. The results are compared to each "case" that has been defined and the statements following a match are executed. If no matching "case" is found, the "default case" is executed (if it is defined).

Which Case?

Lets take a closer look at the switch statement. The expression can be a single variable, the result of a function call, or any other valid statement. "Case" statements within a switch statement serve as branch labels and must include a CONSTANT value *not* a variable. For example "case 3: " is a valid case statement, while "case varx:" is not valid, since varx is not a constant. Remember, the case must be labeled by a constant value or a constant expression.

A switch statement can contain any number of cases, but no two case constants may have the same value. To cause a branch to occur, the constant value must exactly match the expression's results. The only exception to this rule is the "default" case. (Notice that the "default" label does not actually have the word "case" preceding it.) If a default case is included (it is optional), and the expression does not match one of the cases, control branches there.

It is important to note that the cases are merely labels. Once the label is "jumped" to, control continues to fall through the remainder of the switch statement. (This is analogous to the goto statement and label.) When more than one "case" is expected to execute the same code, this "fall through" condition can be used to your advantage. However, more often you want execution of code to end before following into another case. This is accomplished with the "break" statement.

Gimme a Break!

The break statement serves a similar purpose in the switch statement and within loops. When a "break" is encountered, program control jumps to the next statement following the switch statement. The break statement is important to proper functioning of the switch statement. To see how the break can be used, look at this example:

```
switch(value){       /* determine what to do with value */

   case 1:           /* if value = 1 then */
     statements      /* do some statements */
     break;          /* case 1 is complete, end switch */

   case 2:           /* if value = 1 then */
     statements      /* do these statements */
     break;          /* case 2 now complete */

   default:          /* if value is some other value */
     statements      /* do these statements */
     break;          /* the default case is complete */
   }
```

Break statements transfer program control once a case is completed. If the break statement is omitted between case 1 and case 2, whenever the value matches the first case, the statements of both case 1 and 2 are executed. Notice that a break statement again occurs after the default case. Although it is not necessary to have a break after the last case in a switch statement (because the next statement to be executed comes after the switch instruction), it is still a good practice. The break may help avoid an error in execution if another case is later added to the end of the switch.

Notice the format used within the switch statement. The syntax diagram shows that the statements can begin anywhere after the colon in a case statement. To make the code more readable, most people prefer to place the case labels on separate lines. Additionally, notice the indentation. Some people like to keep the case statements lined up with the switch statement; others prefer to indent. Almost everyone agrees, though, that the statements should be indented after the case statement.

(continued)

Each case is compared to the value, and the default case is taken only if no match is found. This means the order of the case statements is not important. In our last example, the default case could be at the top (case 1 could follow case 2), and the statement would still function the same way. For easy maintenance and understanding, I recommend that you order the case statements whenever possible, and place the default at the bottom. This set-up makes it easier to determine if every possible branch has been included.

You may notice that I mention little regarding the statements following the cases. These statements can be any valid C instructions, including other switch statements. The only important point to remember is that once execution has begun at a case label, it continues through the remainder of the switch, unless a break statement is encountered (or a goto statement, of course).

Let's Code!

To help us understand the switch statement let's write a program. Here are the parameters this program should follow:

1.) Count the number of times the keystroke '1' is entered.
2.) Count the number of times '2' or '3' or '9' are entered.
3.) Count the number of times any other keystrokes are entered.
4.) Print the results of each count then terminate the program when 'Q' or 'q' are entered.

When you have finished writing the program, it should look something like the program in Listing one. If you have problems, don't be afraid to seek assistance from my example or a language reference manual. After your program works successfully, proceed to expanding or alterations. Remember, you can learn a lot from experimenting.

Listing One

```
/* Program Switch.c */
/* This program will demonstrate the use of the switch */
/* Keyboard input will be accepted and a tally will be kept */
/* for each of the following */
/* 1s that are entered */
/* 2s or 3s or 9s that are entered */
/* all other keys that are entered except Q and q */
/* these will terminate the program */

/* since keyboard input is normally buffered on the */
/* Amiga it will be necessary to press enter before input keystrokes */
/* can be evaluated */


#include <stdio.h>

main()                  /* start of program */
{
    short ones;         /* place to hold ones */
    short two_3_9;      /* place to hold 2s 3s and 9s */
    short others;       /* place to hold others */


    ones = two_3_9 = others = 0;  /* initialize all to 0 */


    printf("Input some keys then press enter \n");
    printf("Press Q or q then enter to end \n");


    for(;;){                    /* forever loop */
        switch(getchar()){      /* get the keystroke to evaluate */
            case 'q':           /* quit key */
            case 'Q':           /* other quit key */
                printf(" 1s = %d, 2s 3s or 9s = %d, others = %d \n",
                            ones,two_3_9,others);
                exit(0);        /* exit will end the program */
                                /* a break statement is not needed */
                                /* since exit ends the program */

            case '1':           /* if a 1 */
                ones++;         /* increment */
                break;          /* end of case */

            case '2':           /* if a 2 */
            case '3':           /* if a 3 */
            case '9':           /* if a 9 */
                two_3_9++;      /* increment */
                break;          /* end of cases */

            default:            /* all others go here */
                others++;       /* increment */
                break;
        }                       /* end of switch statement */
    }                           /* bottom of forever loop */
}                               /* end of program and function main */
```

•AC•

Send any questions or comments to:

The C Group
c/o *Amazing Computing*
P.O. Box 869
Fall River, MA 02722

Easy Menus in JForth

The EZMenu system makes it quite easy to implement simple text based menus

by Phil Burk

Never Used Forth?

If you have already programmed in Forth, skip ahead to the next section!

I won't try to teach you Forth in this article. But hopefully I can explain enough so that the program listing makes some sense. In Forth, subroutine, or function is called a "word". The definition of a word is started with a colon, ':', followed by the name of the word. The definition is terminated with a semicolon, ';'. A program called HI that prints out "Hello World!" would be defined as follows:

```
: HI  ." Hello World!" ;
```

This could be compiled interactively in any Forth in 1 or 2 seconds. Once compiled it is added to a dictionary of commands, any of which can be executed interactively or referenced in another program. Thus, Forth is both a compiler and an interactive environment.

Forth code looks a little strange at first since it uses Reverse Polish Notation. In Forth, numbers and addresses are held on a stack. The Forth words operate on this stack in the order of their appearance in the code. The syntax is therefore very simple, being much like English. Here is some code that places two numbers on the stack, adds them together with "plus", then prints the answer using "dot".

```
23 45 + .
```

The answer, 68, would be printed if you typed this into any Forth.

Most of the common functions in Forth are standardized. However, the way in which Forth is interfaced to an operating system has not been standardized. Thus, this program, which uses Amiga Intuition Menus, will only compile under JForth from Delta Research. The EZMenu toolbox is unique to JForth, but almost every Forth on the Amiga allows you to access Intuition Menus in some way.

Why Use Pull Down Menus?

If you're like me, when you buy a new interactive program you want to try it out NOW! You pop in the disk, click on some likely looking icon and you're in. Now what? While the manual sits unopened in the box you probably start exploring the pull down menus. If the user interface is well designed you can get pretty far this way. After your initial frenzy subsides, and you get in deeper, you can always consult the manual to tell you what you've missed.

If you are writing a program for others, providing a good set of menus is obviously important. Adding menus to your Amiga program, however, is no trivial matter. The Amiga menu system is so flexible, and has so many options, that it can take a lot of work even to do something simple. I am very grateful to the folks at Amiga for providing a very well thought out menu system that can handle almost anything. But when I was just writing a simple text menu, I used to wish for an easier way. For this reason, I wrote the EZMenu system and included it with the JForth compiler. The EZMenu system makes it quite easy to implement simple text based menus which are the most common type. If you choose to do something fancier, like including graphics in your menus, you will have to do a bit more work.

[Note: JForth is a Forth '83 based compiler similar to the MultiForth system described in previous Amazing Computing articles.]

Amiga Intuition Menus

To understand how EZMenu system works, it helps to have some idea of how the Intuition Menu system works. Intuition Menus use several different structures linked together. (See Figure 1. for a diagram of this system.) The primary structure is the Menu structure. This structure determines where the menu appears on the menu bar, its name, and its size. The Menu structure points to a linked list of MenuItems.

Each MenuItem has its own size and position, plus information on how to draw it. Each MenuItem points to either an IntuiText structure or an Image structure. Thus you can mix text and graphic images in a menu. MenuItems also have a number of flags that control whether it has a checkmark, how it is highlighted, whether it has a command key, etc. MenuItems may also have a pointer towards a list of SubItems.

The IntuiText Structure has a pointer to the text, plus information on position, font, colors, etc. Multiple IntuiText structures can be linked together into lists. Image structures allow you to specify a bit mapped image, which bit planes to use for highlighting, size and position, etc. Images can also be linked together.

The primary Menu structures can be linked together to form a Menu Strip which can be connected to a window. When you select a window, its Menu Strip is made available. When you pick from a menu, a MENUPICK event is sent to the window for use by your program.

(continued)

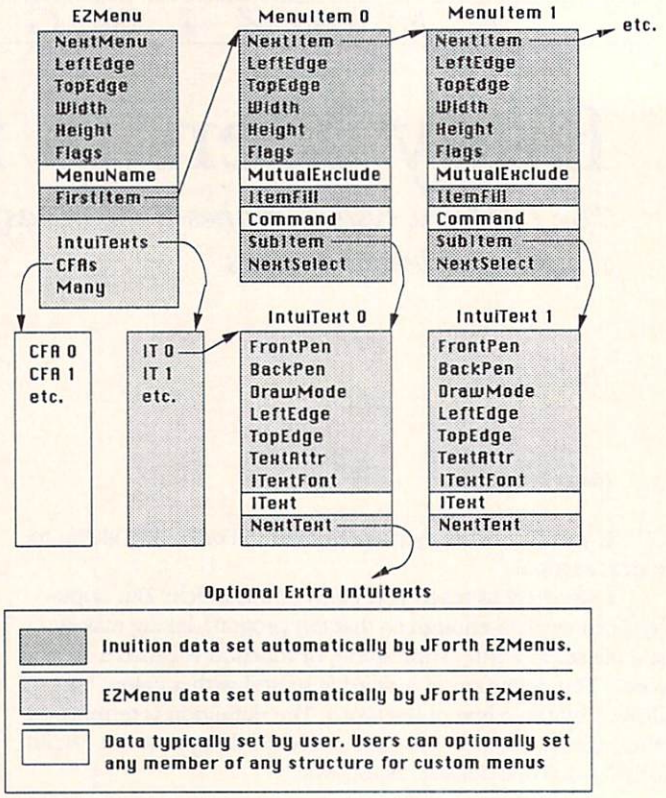


Figure 1

Although complicated, the Intuition Menu system is very powerful and flexible. The JForth EZMenu system was designed to make it easier to use these menus in your programs.

JForth EZMenu System

The JForth EZMenu system makes some assumptions about how you'd like your menus to look. Using these assumptions, it can do most of the work required to set up an Amiga Intuition menu. EZMenu assumes that the MenuItems will be lines of text arranged vertically, "regular menus". You simply specify how many items to have and what the text will be. You can then determine a Forth word to be executed when you pick each item. You may also add command key equivalences, checkmarks with mutual exclusion, and other custom features as desired.

The EZMenu system is based around a special JForth structure called the EZMENU. This structure contains a complete Intuition Menu structure, plus a pointer to a block of IntuiText structures and a pointer to an array of CFAs, one for each menu item, and a count of how many menu items there are. You create one of these structures for each menu wanted. Most of the EZMenu routines uses this structure as one of its parameters.

Let's look at a simple interactive graphics program written in JForth to see how this EZMenu system is used.

Step by Step

This program will open a window and begin drawing colored lines or boxes in a random walk pattern. The pull down menu offers four choices. You can select between lines or boxes, clear the window, or quit.

The numbers in parentheses in this article correspond to numbers in the listing. This should make it easier to match code with the text.

(1) Compile any code needed for this program. This includes the graphics toolbox, the event handling tools, the EZMenu system, and a random number generator.

(2) Declare a copy of an EZMENU structure as described above. We will be using only one menu.

(3) Define the words and variables that control the drawing mode. The variable DRAW-MODE is set by picking Lines or Boxes from the menu. This variable is then used by another routine to decide what to draw.

(4) Define a word to clear the window. This demonstrates how to call an Amiga library routine from JForth. We will use the Amiga SetRast function to set the entire RastPort to the background color. The first line in CLEAR.WINDOW gets the address of the current RastPort. This variable will be set when the window is opened. The routine is called with the line:

```
CALL GRAPHICS_LIB SetRast
```

The CALL word in JForth builds a call to the named routine by searching the Amiga "FD" files for the necessary information. It figures out which parameters go in which 68000 register, determines the offset of the routine in the library, then builds the proper 68000 machine code. This system will work with any Amiga library that has an FD file including the ARP library, custom MIDI libraries, or whatever.

(5) The word MY-MENU.INIT initializes the menu. First we set the width for the menu items to 10 pixels. Then we dynamically allocate the structures needed for our 4 menu items with the line:

```
4 MY-MENU EZMENU.ALLOC
```

The word EZMENU.ALLOC allocates enough memory for 4 MenuItems and 4 IntuiText structures and attaches them to MY-MENU. It also allocates space for 4 CFAs. The next command line uses EZMENU.SETUP to give the menu a name. It also initializes all of the Menu, MenuItem, and IntuiText structures to reasonable defaults, then links these structures together into a complete Intuition menu.

We now use the word EZMENU.TEXT! to specify the text for each menu item. Notice how we use the word 0" to generate the NUL terminated, 'C' like, text strings needed by Intuition. We now use EZMENU.CFA[] to tell the EZMenu system what to do when a menu item is picked. We could set each one individually like we did with the text above, but I decided to use a DO LOOP just for fun. We could stop here and have a workable pull down menu. Let's continue, however, and make them a little fancier.

(6) Put a checkmark beside the Lines or Boxes item in the menu to show which one is current. We can use the Amiga's mutual exclusion feature to make one check mark automatically disappear when the other appears. Intuition allows you to give each menu item a bit pattern. When you select a menu item, its pattern tells Intuition which other menu items to turn off. There are 32 bits in the pattern but we will only look at the 4 lowest bits since we have only four items.

When we select menu item 1, "Boxes", we want menu item 0, "Lines" to become unchecked. The "Boxes" item would now have the checkmark. The other items will be unaffected. Thus the exclusion pattern for Boxes should have bit 0 set to 1. The bits are numbered from right to left, 0-31. Thus the pattern

(continued)

for "Boxes" should be 0001. For "Lines" to turn off "Boxes" the pattern should be 0010. The binary patterns for each menu item are shown below.

```
                        3210  = Item # for each bit
Lines       - bit 0     0010  = turn off item 1, Boxes
Boxes       - bit 1     0001  = turn off item 0, Lines
Clear       - bit 2     0000  = leave others alone
Quit        - bit 3     0000
```

We also need to put a checkmark beside the "Lines" item since the program starts that way.

(7) We can easily assign a command sequence for menu items using the EZMENU.COMMSEQ! word. Now when the program is running you will be able to hold down the "right Amiga key" and then hit a 'C' on the ASCII keyboard to Clear the window. The menu initialization is now complete.

(8) This code is responsible for drawing the lines and boxes. SAFE.RECT sorts the corners of a rectangle before drawing it. The Amiga does not check for backwards rectangles (for speed reasons) and freaks out if you try to draw one. Notice the use of GR.RECT, is one of the JForth graphic routines which start with the prefix "GR". (The JForth graphics toolbox is based on the concept of a current window. Most JForth graphics words operate on the RastPort of that window.)

The word WANDER.XY is interesting because it uses JForth's structure referencing tools. If you have done any Amiga programming, you have no doubt encountered the use of structures. Structures allow you to package together the data needed to describe something in the computer, like a window or menu. In this word, I wanted to keep the lines and boxes from going outside the window. I couldn't just use the original sizes because people might resize the window. Luckily, the current width and height are stored inside the window structure. The x position is clipped to the window by the line:

```
0 MAX GR-CURWINDOW @ ..@ wd_GZZWidth MIN
```

Let's examine this line in detail. The x value has been left on the stack by the previous line. The word MAX takes two numbers off the stack and returns the biggest. Thus if our x position is negative, zero is bigger and we are left with zero. If x is positive, we are left with x. We then get our window structure pointer from the variable GR-CURWINDOW. This is passed to ..@ which fetches the width of the GIMMEZEROZERO window from the structure. In 'C' this line would look something like:

```
if (x < 0) x = 0;  /* equivalent 'C' code! */
xlimit = gr_curwindow->GZZWidth;
if (x > xlimit) x = xlimit;
```

The JForth word ..@ calculates an address by adding the offset for the wdGZZWidth member to the structure. It then calls either C@ , W@ or @ depending on whether the width is an 8 bit, 16 bit or 32 bit value. You don't have to know the size of a structure member to reference it. We have a saying around here, "The size of your member is not as important as what you do with it!". The final word, MIN, clips the x value to the window width. There are more examples of accessing structures in section (10).

(9) This section contains the program's main loop. The word LOOP.DRAW draws a line or box then checks to see if the user has generated any events. The events are associated with a specific window so we must pass EV.GETCLASS our window address. It returns an event class. If the class is zero, there was no event so we just keep looping. If there was an event we pass it to HANDLE.EVENT for processing. HANDLE.EVENT only checks for two kinds of events, MENUPICK and CLOSEWINDOW. If a menu is picked, the menu code is passed to EXMENU.EXEC which then figures out the item selected. EZMENU.EXEC then calls the word you specified in section 5 using EZMENU.CFA[]. If the CLOSEWINDOW gadget was hit, we set QUIT-NOW which causes LOOP.DRAW to exit later.

(10) I find it very helpful to separate most programs into three sections: Initialization, Execution, and Termination. If you look at the final word EZWALKER, you can see this organization. When debugging, I can initialize the application with one word, e.g. EZMENU.INIT. I can then examine structures, test graphic words, look at variables, etc., under the same conditions that the program would execute. When I am done, I can clean up with one word.

At the beginning of this section we declare a NewWindow structure. This is used as a template for how we want our window to look when we open it.. We can set the default values for this window using:

```
MY-WINDOW NewWindow.Setup
```

We can then override these defaults to customize our window. In the next two lines, we can give it our own title by storing the absolute address of a string in the title field of the NewWindow structure. The example in section (8) used a pointer to a structure. Here we use the structure directly. Note also that we use 0" since the Amiga uses zero terminated strings instead of Forth style strings. The word ..! is the opposite of ..@ as it is used to store values in a structure.

We need to change a few of the flags to make menus work with this application. We set the IDCMP flags to give us menu picks and closewindow events using the line:

```
CLOSEWINDOW MENUPICK |
  MY-WINDOW ..! nw_IDCMPFlags
```

The equivalent code in 'C' would be:

```
my_window.IDCMPFlags = CLOSEWINDOW | MENUPICK;
```

By also setting the ACTIVATE flag, we don't have to click in the window to active when it opens. This flag is ORed with the existing flags. We now open the window and associate our menu with it using SetMenuStrip().

The Termination word clears the menu strip, closes the window, then frees any memory associated with the EZMenu. The last word EZWALKER ties everything together. Finally I print a message that tells me how to run the program immediately after compiling.

Conclusion

I hope this article will encourage you to use menus in your application (if you are not already doing so). To get the most out of Amiga's menus you should read the Intuition manual. Menus can make your programs easier to use and help give them a professional look.

I will try to upload this program onto most bulletin boards so you don't have to type it in. It shouldn't take too long to download. The source code is 5215 bytes. The executable image is 9324 bytes and the small image size is because of CLONE: an optimizing target compiler to be released in late 1988.

If you can't find EZWalker on a BBS you can send $5.00 to:
Delta Research, P.O. Box 1051, San Rafael, CA 94915

We will send you a public domain disk containing this program and others. The $5.00 can be applied toward the price of JForth when purchased from Delta Research. More extensive examples are included with the JForth compiler that demonstrate multiple menus, enabling menu items, etc. If you have questions about this program, call me at (415) 485-6867.

Listing One

```
\ Demonstrate the use of JForth's EZMenu system.
\ Use pull down menus in a simple graphics application.
\
\ Author: Phil Burk
\ Delta Research, Box 1051, San Rafael, CA, 94915
\ (415) 485-6867
\ July 8, 1988
\
\ This code is hereby placed in the Public Domain
\ and may be freely distributed.

\ (1) Conditionally compile support code not already loaded
include? newwindow.setup ju:amiga_graph
include? ev.getclass     ju:amiga_events
include? ezmenu ju:amiga_menus
include? choose ju:random

\ Forget this code if already loaded.
ANEW TASK-EZWALKER.F

\ (2) Declare an EZMenu structure.
EZMENU MY-MENU

\ (3) ----------------------
\ Variables used to control application.
variable DRAW-MODE   ( lines or boxes )
0 constant USE_LINES
1 constant USE_BOXES

variable QUIT-NOW    ( time to stop? )
variable LAST-X
variable LAST-Y

\ Define words (functions) to call when menu item picked.
: USE.LINES  ( - , set application drawing mode to lines)
    use_lines draw-mode !
    last-x @ last-y @ gr.move
;

: USE.BOXES  ( - , now draw boxes )
    use_boxes draw-mode !
;

\ (4) Call any Amiga Library routine by name
\ using the JForth CALL facility.
: CLEAR.WINDOW ( - , set rastport to color 0 )
    gr-currport @ ( get absolute addr of window rastport)
    0     ( background color )
```

(continued)

```
    call graphics_lib SetRast   ( call Amiga routine )
    drop   ( don't need return value )
;

: QUIT.DRAWING   ( - , set termination flag )
    quit-now on
;

\ (5) ------------------------
\ Set up Menu and Menu items using EZMENU system.
: MY-MENU.INIT   ( - , initialize menu )
    110 menuitem-defwidth !   ( set default item width )
\ Allocate space for 4 menu items with intuitext structures
    4 my-menu ezmenu.alloc
\
\ Set name of menu and position in list.
    0" Choose" 0 my-menu ezmenu.setup
\
\ Define the text for each menu item.
    0" Lines" 0 my-menu ezmenu.text!
    0" Boxes" 1 my-menu ezmenu.text!
    0" Clear" 2 my-menu ezmenu.text!
    0" Quit"  3 my-menu ezmenu.text!
\
\ Set the function to call for each menu item.
\ Pull off stack in reverse order.
    ' quit.drawing     ' clear.window
    ' use.boxes        ' use.lines
    4 0 DO  i my-menu ezmenu.cfa[] !  LOOP
\
\ (6) Set lines and boxes item to have exclusive checkmarks
    [ BINARY ] ( Use base 2 to express exclusion pattern.)
    0010 0 my-menu ezmenu.exclude!
    0001 1 my-menu ezmenu.exclude!
    CHECKED 0 my-menu ezmenu.set.flag
    [ DECIMAL ]
\
\ (7) Set Command Sequence keys for Clear and Quit.
    ascii C 2 my-menu ezmenu.commseq!
    ascii Q 3 my-menu ezmenu.commseq!
;

\ (8) ------------------------
\ Code for drawing lines and boxes.
: SAFE.RECT ( x1 y1 x2 y2 - , sort corners and draw )
    >r swap >r 2sort   ( sort X values )
    r> r> -2sort       ( sort Y values )
    -rot      ( - x y x y )
    gr.rect
;

: DRAW.NEW.XY ( x y - , draw either a line or a box )
    draw-mode @
    use_lines =
    IF  2dup gr.draw
    ELSE
        2dup last-x @ last-y @ safe.rect
    THEN
    last-y ! last-x !
;

: NEXT.COLOR ( - , Cycle through colors 1,2,3 )
    gr.color@ 1+ dup 3 >
    IF drop 1
    THEN  gr.color!
;

\ Select random distances for random walk.
: CALC.DELTA.X ( - dx )
    41 choose 20 -
;
: CALC.DELTA.Y ( - dy )
    21 choose 10 -
;

: WANDER.XY ( - , random walk )
\ Add a number between -20 and +20
    last-x @ calc.delta.x +
\ Clip to 0 and current window size.
\ Note reference to window structure.
    0 max gr-curwindow @ ..@ wd_gzzwidth min
\
    last-y @ calc.delta.y +
    0 max gr-curwindow @ ..@ wd_gzzheight min
    draw.new.xy
;

\ (9) ------------------------
\ Process IDCMP events.
: HANDLE.EVENT ( event_class - )
    CASE
\ Call functions set using EXMENU.CFA[] !
        MENUPICK
        OF  ev-last-code @ my-menu ezmenu.exec
        ENDOF
\
\ Set quit flag if CLOSEBOX hit.
        CLOSEWINDOW
        OF quit-now on
        ENDOF
\
        ." Unrecognized event!" cr
    ENDCASE
;

\ Draw lines or boxes until told to quit.
: LOOP.DRAW ( - )
    quit-now off
    BEGIN
        wander.xy   ( do graphics )
        next.color
\
        gr-curwindow @ ev.getclass ?dup
        IF handle.event
        THEN
        quit-now @
    UNTIL
;

\ Declare new window structure.
NewWindow MY-WINDOW

: EZWALKER.INIT ( - , set everything up )
    gr.init   ( initialize graphics system )
    my-window newwindow.setup   ( set defaults )
\
\ Change window title using structure access word ..!
    0" EZWAlker  in JForth  by Phil Burk" >abs
    my-window ..! nw_Title
\
\ Set flags for window to allow menus.
    CLOSEWINDOW MENUPICK |
    my-window ..! nw_IDCMPFlags
\ Make window automatically active.
    MY-WINDOW ..@ nw_Flags   ACTIVATE |
    MY-WINDOW ..! nw_Flags
\
\ Open window based on NewWindow template
    my-window gr.openwindow gr.set.curwindow
\ Initialize menu and attach to window.
    my-menu.init
    gr-curwindow @ my-menu SetMenuStrip()
\ Start in middle of window
    gr_xmax 2/ last-x !
    gr_ymax 2/ last-y !
    use.lines
;
: EZWALKER.TERM ( - , clean up menus and close window. )
    gr-curwindow @ ClearMenuStrip()
    gr.closecurw
    my-menu ezmenu.free
;
: EZWALKER ( - , do it all)
    EZWALKER.init
    loop.draw
    EZWALKER.term
;

cr ." Enter:    EZWALKER    to see demo." cr
```

•AC•

Getting Started in Assembly

Learn to write the software in the most efficient computer language available: ***Assembly***

by Jeff Glatt
BIX:jfiore

The surest way to make a powerful computer an impotent collection of deficient electronics is to run a program written in an inefficient programming language. If the application involves intricate calculations such as floating point math or needs to perform a real-time function such as music sequencing or animation, a high level language just won't work. The only solution is to write the software in the most efficient computer language available: Assembly.

Although C has become the language of choice for many programmers, and the de facto standard for the Amiga, it is certainly incapable of matching assembly's speed performance. In applications where unnecessary, extra machine instructions can cause a humanly perceptible delay, a high level language like C can ruin a perfectly good algorithm.

Unfortunately, there is precious little information about programming in assembly on the Amiga. Now I wish to present an example implementing some very elementary functions. Although there is no urgent need for speed in executing these functions, they demonstrate, in assembly, how to perform the following aspects of Amiga programming:

1) Making a program that runs from the CLI or WorkBench.
2) Opening and calling routines in Amiga libraries.
3) Opening windows and fonts.
4) Setting up and decoding menus.
5) Getting and interpreting messages from Intuition including keyboard and mouse events, and gadget selections.
6) Outputting text and changing pen colors.

If you can't do the preceding functions in the language of your choice, then you aren't really writing Amiga software.

The first step in programming in assembly on the Amiga is to become familiar with the Motorola 68000's instruction set. If you don't know what a move instruction is then proceed no further. You need to study a book on the 68000. I recommend "68000 Assembly Language Programming" by Leventhal, McGraw-Hill publications.

Next, purchase an assembler. Amazing Computing, June 1988, contained a review of several products. The assembler should come with some include files which define certain structures and values that we need to reference.

The disc should contain a start-up code module as described in the RKM Libraries and Devices manual on page 489. You can obtain this start-up code on Fish Discs 101 or 55. By calling the first program routine to be executed _main, and linking with this start-up code, you will automatically achieve the first goal: a program executable from the CLI or WorkBench (provided that you make an Icon for WorkBench). Also, the start-up code gets the Exec library's base, called _SysBase, and opens the DOS library with the base address stored at _DOSBase.

In describing the example assembly program, Example1, I will step through the code in basically the same order the 68000 executes the instructions, referring to the symbolic labels at certain lines. At the top of the listing are some external references that the include files with the assembler will resolve. These include files supply the actual values for the symbolic names.

The first line to be executed is E1 in _main. The contents of all of the non-scratch registers that will be used are saved here. The registers d0, d1, a0, and a1 are considered scratch. This means these registers can be used without saving the original contents. All other registers must be saved if they are to be used. Never violate this rule unless you prominently comment, at the head of the routine, which non-scratch registers will be "destroyed". If a call is made to an Amiga library function which uses a non-scratch register, save the register before the call, and restore it upon return. At line E2, a call is made to the routine, open_libs.

The routine, open_libs, opens the libraries, font, and window the program will access, attaches the menu, and sets the initial drawing mode and primary pen color. If all goes well, it returns to _main with d0 = 1. If anything fails to work, it is set up to return d0 = 0. If you link with the start-up code, the Exec and DOS libraries have already been opened, and the base addresses stored at _SysBase and _DOSBase respectively. When linking, always specify the start-up code first so that it calls the application's _main routine. A library must be opened before accessing the routines in it. Whenever a library routine is called, the base address of that routine's library must be in a6.

To open the Intuition library (as shown at line B1), we must call the OpenLibrary routine which is itself a routine in the Exec library. That is why _SysBase is placed in a6. The return value (in d0) will be the base address of the Intuition library, or 0 if the Intuition library doesn't open.

Many Amiga functions return certain values for an error. An alphabetical listing of available library routines starts on page A-12 of the RKM Libraries and Devices. It tells what parameters must be placed in which registers, and how to interpret any return value. If OpenLibrary returns d0 = 0 (error), then storing the value at _IntuitionBase will set the Z flag, and we will branch to B10 before even proceeding to B2. (There is no need for a tst instruction here. In fact, moving a value into another

(continued)

register is quicker than using the tst instruction to set the flags in the status register. Just be careful that the register moved to is a data, not address, register. Moving to an address register automatically uses the movea instruction which doesn't affect ANY flags.)

At B2, the Graphics library is opened just like Intuition except that _SysBase needn't be placed back in a6. It's still there since the Amiga libraries never destroy a6. The address of the main task is obtained at B3. Although this particular program doesn't use this info, any program that uses multi-tasking and ports probably will. Since future articles will demonstrate how to add these features to the Example1 skeleton code, we'll get the address now.

At B4, the TextFont structure's address, TextAttr, is passed to a routine called OpenFont in the Graphics library. If you examine the TextAttr structure in the data section of the program, you will note that I want to open the Topaz 8 Font. By changing the FontName string to 'sapphire.font', and the ySize in TextAttr, I could open a different font. Also I can change the FC_STYLE to UNDERLINED (1), BOLD (2), ITALIC (4), or EXTENDED (8).

At B5, the main window is opened via a call to Intuition lib. One of the passed parameters is the base of the newWindow structure. This structure, called WindowStruct, is in the data section of the program, and it contains info about where the window will be placed, how big it will be, what pen colors to use for the foreground and background, which types of messages Intuition will send to the window, what type of screen the window will be opened upon, what should appear in the title bar, what types of system gadgets will be in the window, a pointer to the list of gadgets that you wish to add to the window, as well as info about sizing dimensions.

Page D-155 of RKM presents a summation of the various fields. The IDCMPflags are particularly important to the program that needs to communicate with Intuition. Page D-152 of the RKM contains list of the values for all possible messages that can be sent. Here are the messages that Example1 wants Intuition to send, the values for the messages, and the bit #s which are set by these values.

Flag	Value	Bit #	When Intuition sends the message
MOUSEBUTTONS	8	3	With every right mouse button press, or release.
MOUSEMOVE	$10	4	Every time the mouse moves.
GADGETDOWN	$20	5	When the right mouse button is depressed over a (user) gadget.
GADGETUP	$40	6	When the right mouse button is released after selecting a gadget.
MENUPICK	$100	8	When a menu selection is made. (not simply when the menu is displayed).
CLOSEWINDOW	$200	9	When the close gadget in the top left corner is selected. (the WINDOWCLOSE flag of Window flags must be set).
RAWKEY	$400	10	When a key is pressed or released.
TOTAL	$778		

Simply add up the values for the desired flags and put this value in the IDCMPflags field of the window structure. (Remember to convert hex values to decimal before adding them). Now when the window is opened, Intuition sets up a port for the window which is where these messages will be sent. Intuition also gets a particular signal which it associates with the "_main" task. This will enable the task to wait for messages to arrive at this port from Intuition, and when the message does arrive, exec will send the program the signal Intuition has allocated.

Many Graphics library routines require the address of the window's RastPort. This address can be found at an offset of 50 bytes from the base of the opened window. The window's RastPort address is obtained at B6.

At B7, the opened font, topaz 8, is set for this window. This means any text output to the window will be rendered with this font. Also, in the menu structures (and any user gadget structures) I have deliberately set any pointers to a TextFont structure, to TextAttr in the data section. It is not necessary to open and set a font because the default font in Preferences will be used, but it is a good idea to do so. Especially with gadgets, if text is to be "fit inbetween" two images, and the co-ordinates are set based on an 80 column display. When the program is run on a 60 column display, the larger letters may trash some of the adjacent image. When you open a specific font, you get exactly the size and style desired.

At B8, the pre-initialized menu strip is added to the window. This program contains 3 menus, each menu containing 3 items. An item in the first menu, Color, will also have 3 subitems. A menu strip is a long, linked list of structures containing Menu, MenuItem, and Intuitext structures. The data section contains examples of each structure. Menu and MenuItem structures contain info about the item or menu's placement (always relative to the top left corner), the dimensions of the select box (the box which is drawn around the item that the mouse pointer is over), the address of the next member of the list, the address of any IntuiText or Image structure (the text or

image that appears in the menu), and a bit mask describing which other items need to be excluded when this one is selected. A flags field describes what features are to be used for this item. A list of possible features for menu items is available on page D-146 of the RKM.

For example, in the Pen1SubItem structure, these features are set: CHECKIT (1), ITEMTEXT (2), COMMSEQ (4), ITEMENABLED ($10), HIGHCOMP ($40), and CHECKED ($100). These 6 values are summed, and this sum is placed in the flags field of the structure. One of the flags set is COMMSEQ. This gives the alternate keyboard command which is indicated in the menu by the symbol for the right Amiga key followed by the chosen ascii value. I chose to use the '1' key as an alternate keyboard shortcut for Pen1, and so the ascii value for '1' must be placed in the Command field of the structure.

There are several different methods for highlighting the select box. For most items, the familiar HIGHCOMP flag has been set, but for one item, InfoItem, HIGHBOX is set instead.

Note that in the menu, a box is drawn around the item as opposed to the entire area being complemented. There will be 3 subitems for the Color item: Pen1, Pen2, Pen3. The selected pen should have a check drawn before it (CHECKIT), and initially, Pen1 must be CHECKED. Only one pen may be selected at a time. The other two must be deselected and unchecked. Do this by mutually excluding each pen from the other two pens.

The mutual exclude field is a bit map of excluded items. Notice that for the Pen1SubItem, all the bits of the mask (except for bit #0 because this is SubItem #0 under the Color Item) are set. This means that when Pen1 is selected, all the other SubItems (Pen2 and Pen3) will be deselected. Notice also that the mutual exclude fields of Pen2 and Pen3 have the appropriate bits set to exclude the other pens.

IntuiText structures are used for every text string Intuition must display. They are used for text with gadgets, menus, and requesters. The various fields give info about where the text should be printed, what colors to use for the fore and background, the drawing mode, and the addresses of the actual string to be printed, the TextFont structure, and any other IntuiText structures linked to this one. An example of this is InfoText.

Once the menu has been attached to the window, the drawing mode and pen colors are initially set at line B9. The drawing mode is JAM2 which means that 2 pen colors are to be used for rendering text. Pen A is set to the color of color register #1. This pen draws the actual "outline of the letter to be rendered". Pen B is set to color register #0. This pen draws the "background beneath the letter". Since this program opens its window on the WorkBench screen, we can use only 4 of the possible 32 color registers—registers 0 to 3. The background pen is always going to be register 0. (The decision is yours It can be changed at any time). The color menu subitems will allow Pen A to be set to register 1, 2, or 3. This is what the 3 subitems, Pen1, Pen2, and Pen3 are for. It allows the letters that are printed to the screen to be one of 3 different colors.

The last operation open_libs performs is to move a 1 into d0. This clears the Z flag so that _main will know that everything went ok. Upon returning to _main, the low byte of d0 is moved into d1. (Again, this is quicker than a tst instruction). If a zero, something went wrong and the code immediately branches to a call to quit_all. Quit_all closes everything that managed to open in open_libs. Always close anything that

was opened before you exit a program. This includes libraries, fonts, devices, windows, screens, spawned tasks, and deallocate any allocated memory.

If open_libs was successful, the next step is to find the signal number that Intuition allocated in OpenWindow. This signal notifies the task of an arriving message. This signal is stored in the message port that Intuition also allocated and attached to the opened window. At E3, the base of the open window is placed into a0. Then the address of the message port is obtained at an offset of 86 bytes. The signal number has been stored at an offset of 15 bytes. When the signal number is obtained (its a number from 0 to 31), its respective bit # of register d7 is set. This is essentially a bit mask of the signal.

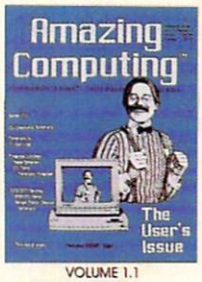
At E4, a call is made to GetMsg to determine if there are any messages at the port. Perhaps the user was moving the mouse while open_libs was executing. (Never trust a user to sit quietly by during critical operations). If there is a message, its address is returned in d0. If no messages, d0 = 0. If a message exists, the code branches to E7. The message contains info about the class of message (mousemove, gadget, etc), as well as other info that can be interpreted different ways depending upon the class. Intuition uses the first 20 bytes of the message structure to keep track of the messages. The info we need starts at the class field, so its effective address is loaded into a0 and post-decrement addressing mode is used to jam the other, following fields into registers.

At this point, it should be noted that normally, indexed addressing mode is used in accessing Amiga system structures, and symbolic labels are used to represent the offsets. For

(continued)

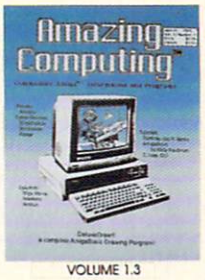
Amazing COMPUTING™

• E X P A N D I N G R E F E R E N C E •



VOLUME 1.1

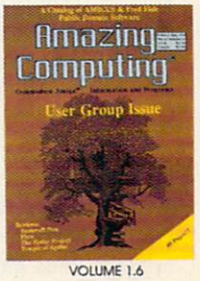
VOLUME 1.2



VOLUME 1.3

VOLUME 1.4

VOLUME 1.5

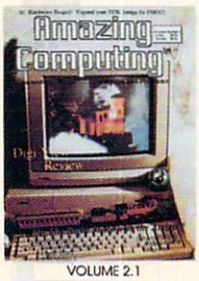


VOLUME 1.6

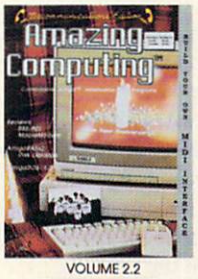
VOLUME 1.7

VOLUME 1.8

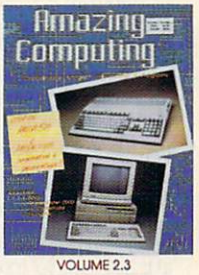
VOLUME 1.9



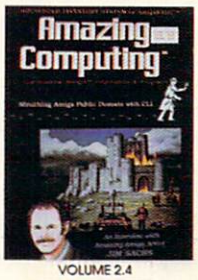
VOLUME 2.1



VOLUME 2.2

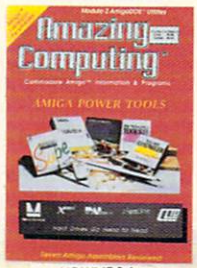


VOLUME 2.3

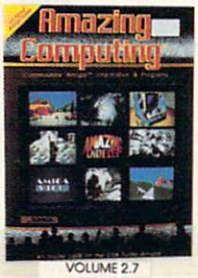


VOLUME 2.4

VOLUME 2.5



VOLUME 2.6

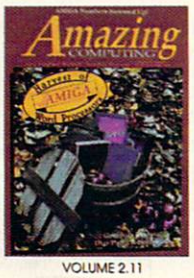


VOLUME 2.7

VOLUME 2.8

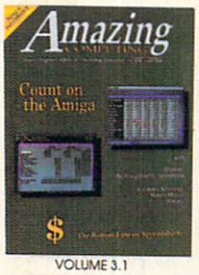
VOLUME 2.9

VOLUME 2.10

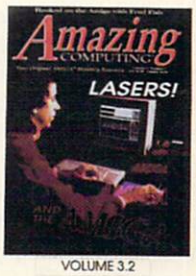


VOLUME 2.11

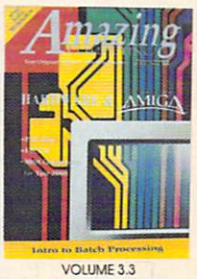
VOLUME 2.12



VOLUME 3.1



VOLUME 3.2



VOLUME 3.3

VOLUME 3.4

VOLUME 3.5

VOLUME 3.6

VOLUME 3.7

VOLUME 3.8

VOLUME 3.9

Your Resource to the Commodore Amiga™

Expanding reference is not just an empty promise. The pages of Amazing Computing™ are filled with articles on technical operations and procedures, basic use, and just-plain-fun. The growing library of Amazing Computing's Back Issues contains articles ranging from building your own IBM Disk controller, to setting up your own startup sequence. Amazing Computing™ has repeatedly been the first magazine to offer the Amiga users solid, in depth reviews and hands on articles for their machines.

From the Beginning

Since February 1986, Amazing Computing™ has been providing users with complete information for their Amigas. This store house of programs and information is still available through our back issues. From the Premiere issue to the present, there are insights into the Amiga any user will find useful. AC was the first magazine to document CLI, tell its readers how to connect a 5 1/4 IBM drive, describe a 1 meg upgrade hardware project for the A1000, and many more. Please read the list of topics AC has covered below to find the information you have been missing.

Back Issues are $5.00 US, $6.00 Canada and Mexico, $7.00 Foreign Surface

All payments must be made by check or money order in U.S. funds drawn on a U.S. Bank.

Limited Supply

Unfortunately, nothing lasts forever, and the availability of some of our Back Issues is definitely limited. Complete your Amazing Computing™ library today, while these issues are still available, by completing the order form in the back of this issue.

Volume 1 Number 1 Premiere 1986

Super Spheres By Kelly Kauffman An ABasic Graphics prog.
Date Virus By J Foust A disease may attack your Amiga!
EZ-Term by Kelly Kauffman An ABasic Terminal program
Miga Mania by P. Kivolowitz Programming fixes & mouse care
Inside CLI by G. Musser a guided insight into the AmigaDos™
CLI Summary by G. Musser Jr. A list of CLI commands
AmigaForum by B. Lubkin Visit Compuserve's Amiga SIG
Commodore Amiga Development Program by D. Hicks
Amiga Products A listing of present and expected products

Volume 1 Number 2 March 1986

Electronic Arts Comes Through A review of software from EA
Inside CLI: part two G. Musser Investigates CLI & ED
A Summary of ED Commands
Live! by Rich Miner A review of the Beta version of Live!
Online and the CTS Fabite 2424 ADH Modem by J. Foust
Superterm V 1.0 By K. Kauffman A term. prog. in Amiga Basic
A Workbench "More" Program by Rick Wirch
Amiga BBS numbers

Volume 1 Number 3 April 1986

Analyze! a review by Ernest Viverios
Reviews of Racter, Barataccas and Mindshadow
Forth! The first of our on-going tutorial
Deluxe Draw!! by R. Wirch An Amiga Basic art program
Amiga Basic, A beginners tutorial
Inside CLI: part 3 by George Musser George gives us PIPE

Volume 1 Number 4 May 1986

SkyFox and Articfox Reviewed
Build your own 5 1/4 Drive Connector By Ernest Viveiros
Amiga Basic Tips by Rich Wirch
Scrimper Part One by P. Kivolowitz prog to print Amiga screen
Microsoft CD ROM Conference by Jim O'Keane
Amiga BBS Numbers

Volume 1 Number 5 1986

The HSI to RGB Conversion Tool
by S. Pietrowicz Color manipulation in BASIC
AmigaNotes by Rick Rae The first of the Amiga music columns
Sidecar A First Look by John Foust A first "under the hood"
John Foust Talks with R. J. Mical at COMDEX™
How does Sidecar affect the Transformer
an interview with Douglas Wyman of Simile
The Commodore Layoffs by J. Foust A look Commodore "cuts"
Scrimper Part Two by Perry Kivolowitz
Marauder reviewed by Rick Wirch
Building Tools by Daniel Kary

Volume 1 Number 6 1986

Temple of Apshai Triology reviewd by Stephen Pietrowicz
The Halley Project: A Mission reviewed by S Pietrowicz
Flow: reviewed by Erv Bobo
Textcraft Plus a First Look by Joe Lowery
How to start your own Amiga User Group by William Simpson
Amiga User Groups
Mailing List by Kelly Kauffman a basic mail list program
Pointer Image Editor by Stephen Pietrowicz
Scrimper: part three by Perry Kivolowitz
Fun With the Amiga Disk Controller by Thom Sterling
Optimize Your AmigaBasic Programs for Speed by Pietrowicz

Volume 1 Number 7 1986

Aegis Draw: CAD comes to the Amiga by Kelly Adams
Try 3D by Jim Meadows an introduction to 3D graphics
Aegis Images/ Animator: a review by Erv Bobo
Deluxe Video Construction Set reviewed by Joe Lowery
Window requesters in Amiga Basic by Steve Michel
ROT by Colin French a 3D graphics editor
"I C What I Think" Ron Peterson with a few C graphic progs
Your Menu Str! by B Catley program Amiga Basic menues
IFF Brush to AmigaBasic 'BOB' Basic editor by M Swinger
Linking C Programs with Assembler Routines...by Gerald Hull

Volume 1 Number 8 1986

The University Amiga By G.Gamble Amiga at Washington State
MicroEd a look at a one man army for the Amiga
MicroEd, The Lewis and Clark Expedition reviewed Frizelle
Scribble Version 2.0 a review
Computers in the Classroom by Robert Frizelle
Two for Study by Frizelle Discovery & TheTalking Coloring Book
True Basic reviewed by Brad Grier
Using your printer with the Amiga
Marble Madness reviewed by Stephen Pietrowicz
Using Fonts from AmigaBasic by Tim Jones
Screen SaVer by P. Kivolowitz A monitor protection prog. in C
Lattice MAKE Utility reviewed by Scott P. Evernden
A Tale of Three EMACS by Steve Poling
.bmap File Reader in Amiga Basic by T Jones

Volume 1 Number 9 1986

Instant Music Reviewed by Steve Pietrowicz
Mindwalker Reviewed by Richard Knepper
The Alegra Memory Board Reviewed by Rich Wirch
TxEd Reviewed by Jan and Cliff Kent
Amazing Directory A guide to the sources and resources
Amiga Developers A listing of Suppliers and Developers
Public Domain Catalog A listing of Amicus and Fred Fish PDS
Dos 2 Dos review R. Knepper Transfer files from PC/MS-DOS
MaxiPlan review by Richard Knepper The Amiga Spreadsheet
Gizmoz by reviewed by Peter Wayner Amiga extras!
The Loan Information Program by Brian Catley
basic prog. to for your financial options
Starting Your Own Amiga Related Business by W. Simpson
Keep Track of Your Business Usage for Taxes by J. Kummer
The Absoft Amiga Fortran Compiler reviewed by R A. Reale
Using Fonts from AmigaBasic, Part Two by Tim Jones
68000 Macros on the Amiga by G. Hull Advance your ability.
TDI Modla-2 Amiga Compiler review by S Faiwisze

Volume 2 Number 1 1987

What Digi-View Is... Or, What Genlock Should Be! by J. Foust
AmigaBasic Default Colors by Bryan Catley
AmigaBasic Titles by Bryan Catley
A Public Domain Modula-2 System reviewed by Warren Block
One Drive Compile by Douglas Lovell Lattice C with one drive
A Megabyte Without Megabucks by Chris Irving
An Internal Megabyte upgrade
Digi-View reviewed by Ed Jakober
Defender of the Crown reviewed by Keith Conforti
Leader Board reviewed by Chuck Raudonis
Roundhill Computer System's PANEL reviewed by Ray Lance
Digi-Paint.....by New Tek previewed by John Foust
Deluxe Paint II .....from Electronic Arts previewed by J. Foust

Volume 2 Number 2 1987

The Modem by Josph L. Rothman efforts of a BBS Sysop
MacroModem reviewed by Stephen R. Pietrowicz
GEMINI or "it takes two to Tango" by Jim Meadows
Gaming between machines
BBS-PCI reviewed by Stephen R. Pietrowicz
The Trouble with Xmodem by Joseph L. Rothman
The ACO Project....Graphic Teleconferencing on the Amiga
by S. R. Pietrowicz
Flight Simulator II....A Cros Country Tutorial by John Rafferty
A Disk Librarian in AmigaBASIC by John Kennan
Creating and Using Amiga Workbench Icons by C. Hansel
AmigaDOS version 1.2 by Clifford Kent
The Amazing MIDI Interface build your own by Richard Rae
AmigaDOS Operating System Calls and
Disk File Management by D. Haynie
Working with the Workbench by Louis A. Mamakos Prog in C

Volume 2 Number 3

The Amiga 2000™ by J Foust
A First look at the new, high end Amiga™
The Amiga 500™ by John Foust
A look at the new, low priced Amiga
An Analysis of the New Amiga PCs by J. Foust
Speculation on the New Amigas
Gemini Part II by Jim Meadows
The concluding article on two-player games
Subscripts and Superscripts in AmigaBASIC by Ivan C. Smith
The Winter Consumer Electronics Show by John Foust
AmigaTrix by W. Block Amiga™ shortcuts
Intuition Gadgets by Harriet Maybeck Tolly
A journey through gadget-land, using C
Shanghai reviewed by Keith M. Conforti
Chessmaster 2000 & Chessmate reviewedby Edwin V. Apel, Jr.
Zing! from Meridian Software reviewed by Ed Bercovitz
Forth! by Jon Bryan Get stereo sound into your Forth programs.
Assembly Language on the Amiga™ by Chris Martin
Roomers by theBandito Genlocks are finally shipping, & MORE!!!
AmigaNotes by R. Rae Hum Busters... "No stereo? Y not?...
The AMICUS Network by J. Foust
CES, user group issues and Amiga Expo"

Volume 2 Number 4 1987

Amazing Interviews Jim Sachs by S. Hull Amiga Artist
The Mouse That Got Restored by Jerry Hull and Bob Rhode
Sluething Public Domain Disks with CLI by John Foust
Highlights: the San Francisco Commodore Show by S Hull
Speaker Sessions: San Francisco Commodore Show H Tolly
Household Inventory System in AmigaBASIC™ by B Catley
Secrets of Screen Dumps by Natkun Okun
Using Function Keys with MicroEmacs by Greg Douglas
Amigatrix II by Warren Block More Amiga shortcuts
Basic Gadgets by Brian Catley Create gadget functions
Gridiron reviewed by K. Conforti Real football for the Amiga
Star Fleet I Version 2.1 reviewed by J. Tracy Amigain Space
The TIC reviewed by J. Foust Battery powered Clock Calendar
Metascope review by H. Tolly An easy-to-use debugger

Volume 2 Number 5 1987

The Perfect Sound Digitizer review by R. Battle
The Future Sound Digitizer by W. Block Applied Vision's SD
Forth! by J. Bryancomparing JForth amd Multi-Forth.
Basic Input by B. Catley AmigaBASIC input routine for use in
all your programs.
Writing a SoundScape Module in C by T. Fay Programming with
MIDI, Amiga and SoundScape by SoundScape author.

Volume 2 Number 5 1987 continued

Programming in 68000 Assembly Language by C. Martin
Continuing with Counters & Addressing Modes.
Using FutureSound with AmigaBASIC by J. Meadows
AmigaBASIC Programming utility with real, digitized STEREO
AmigaNotes Rich Rae reviews SoundScape Sound Sampler.
More AmigaNotes by R. Rae A furtherlock at Perfect Sound .
Waveform Workshop in AmigaBASIC by J. Shields edit & save
waveform for use in other AmigaBASIC programs.
The Mimetics Pro MIDI Studio by Sullivan, Jeffery
A review of Mimetics' music editor/player.
Intuition Gadgets Part II by H. MaybeckTolly Boolean gadgets
provide the user with an on/off user interface.

Volume 2 Number 6 1987

Forth! by J. Bryan Access resources in the ROM Kernal.
The Amazing Computing Hard Disk Review by J. Foust & S. Leemon In-depth looks at the C Ltd. Hard Drive, Microbotics' MAS-Drive20, Byte by Byte's PAL Jr., Supra's 4x4 Hard Drive and Xebec's 9720H Hard Drive. Also. a look at disk driver software currently under devlopment.

Modula-2 AmigaDOS™ Utilities by S. Faiwiszewsk A
Calls to AmigaDOS and the ROM kernal.
Amiga Expansion Peripheral by J. Foust
Explanation of Amiga expansion peripherals.
Amiga Technical Support by J. Foust
How and where to get Amiga tech support.
Goodbye Los Gatos by J. Foust Closing Los Gatos.
The Amicus Network by J. Foust West Coast Computer Faire.
Metacomco Shell and Toolkit by J. Foust A review
The Magic Sac by J. Foust Run Mac programs on your Amiga.
What You Should Know Before Choosing an Amiga 1000
Expansion Device by S. Grant
7 Assemblers for the Amiga by G. Hull Choose your assembler
Shakeup Replaces Top Management at Commodore by S. Hull
Peter J. Baczorby S. Hull Manager at CBM gives an inside look
Logistix A review by Richard Knepper
Organize! by A review Richard Knepper database.
68000 Assembly Language Programming on the Amiga
by Chris Martin
Superbase Personal Relational Database by Ray McCabe
AmigaNotes by Rae, Richard A look at FutureSound
Commodore Shows the Amiga 2000 and 500 at the Boston
Computer Society by H Maybeck Tolly

Volume 2, Number 7 1987

New Breed of Video Products by John Foust...
Very Vivid! by Tim Grantham...
Video and Your Amiga by Oran Sands III
Amigas & Weather Forecasting by Brenden Larson
A-Squared and the Live! Video Digitizer by John Foust.
Aegis Animator Scripts and Cel Animation by John Foust
Quality Video from a Quality Computer by Oran Sands III.
Is IFF Really a Standard? by John Foust..
Amazing Stories and the Amiga™ by John Foust.
All about Printer Drivers by Richard Bielak
Intuition Gadgets by Harriet Maybeck Tolley.
Deluxe Video 1.2 by Bob Eller
Pro Video CG1 by Oran Sands III.
Digi-View 2.0 Digitizer/Software by Jennifer M. Janik
Prism HAM Editor from Impulse by Jennifer M. Janik
Easyl drawing tablet by John Foust..
CSA's Turbo-Amiga Tower by Alfred Aburto
68000 Assembly Language by Chris Martin.

Volume 2, Number 8 1987

This month Amazing Computing™ focuses on entertainment packages for the Amiga. Amazing game reviews...

SDI, Earl Weaver Baseball, Portal, The Surgeon, Little Computer People, Sinbad, StarGlider, King's Quest I,II and III, Faery Tale Adventure, Ultima III, Facets of Adventure, Video Vegas and Bard's Tale.

Plus Amazing monthly columns... Amiga Notes, Roomers, Modula-2, 68000 Assembly Language and The Amicus Network.
Disk-2-Disk by Matthew Leeds
The ColorFonts Standard by John Foust
Skinny C Programs by Robert Riemersma, Jr.
Hidden Messages In Your Amiga™ by John Foust
The Consumer Electronics Show and Comdex by J Foust

Volume 2 Number 9 1987

Analyze 2.0 reviewed by Kim Schaffer
Impact Business Graphics review by Chuck Raudonis
Microfiche Filer review by Harv Laser
Pagesetter review by Rick Wirch
Gizmoz Productivity Set 2.0 review by Bob Eller
Kickwork review by Harv Laser
Diga Telecommunications Package review by Steve Hull
Mouse Time and Timesaver review by John Foust
Insider Memory Expansion review by James O'Keane
Microbotics Starboard-2 review by S. Falwiszewski
Leather Goddesss of Phobos by Harriet Maybeck-Tolly
Lattice C Compiler Version 3.10 reviewed by Gary Sarff
Manx 3.4a Update reviewed by John Foust
AC-BASIC reviewed by Sheldon Leemon
AC-BASIC Compiler an alternative comparison by B Catley
Modula-2 Programming S Falwiszewski Raw Console Dev. Events
Directory Listings Under AmigaDOS by Dave Haynie
AmigaBASIC Patterns by Brian Catley
Programming with Soundscape Todor Fay manipulate's samples
Bill Volk, Vice-President Aegis Development, by Steve Hull
Jim Goodnow, Developer of Manx 'C' interview by Harriet M Tolly
Plus a great collection of monthly columns...

Volume 2 Number 10 1987

Max Headroom and the Amiga by John Foust
Taking the Perfect Screen Shot by Keith Conforti
Amiga Artist: Brian Williams by John Foust
Amiga Forum on Compuserve™... Software Publishing Conference Transcript by Richard Rae
All About Online Conferencing by Richard Rae
dBMAN reviewed by Clifford Kent
Amiga Pascal reviewed by Michael McNeil
AC-BASIC Compiler reviewed by Bryan Catley
68000 Assembly Language by Chris Martin
Amiga Programming:
Amiga BASIC Structures by Steve Michel
Quick and Dirty Bobs by Michael Swinger
Directory Listings Under Amiga-DOS, Part II by Dave Haynie
Fast File I/O with Modula-2 by Steve Falwiszewski
Window I/O by Read Predmore
Plus a great collection of monthly columns...

Volume 2 Number 11 1987

Word Processors Rundown by Geoff Gamble
ProWrite, Scribble!, and WordPerfect compared
LPD Writer Review by Marion Deland
VizaWrite Review by Harv Laser
Aedit Review by Warren Block
WordPerfect Preview by Harv Laser
Jez San Interview by Ed Bercovitz—StarGlider author speaks!
Do-It-yourself Improvements to the Amiga Genlock
Digi-Paint Review by Harv Laser
Sculpt 3D Review by Steve Pietrowicz
Shadowgate Review by Linda Kaplan
TeleGames Review by Michael T. Cabral
Reason Preview: an intense grammar examination application
As I See It by Eddie Churchill WordPerfect,Gizmoz V2.0 and Zing!
AmigaNotes by R Rae 4 electronic music books
Modula-2 Programming by S.Falwiszewski devices, I/O, &serial port
68000 Assembly Language by Chris Martin Display routines
The AMICUS Network by John Foust—Desktop Publishing, Seybold
C Animation Part II by Mike Swinger Animation Objects
BASIC Text by Brian Catley Pixel perfect text positioning
Soundscape Part III by Todor Fay VU Meter and more
Fun with Amiga Numbers by Alan Barnett
File Browser by Bryan Catley—Full Feature BASIC File Browsing
Plus a great collection of monthly columns...

Volume 2 Number 12 1987

The Ultimate Video Accessory by **Larry White**
The Sony Connection by **Stewart Cobb**
15-Puzzle in AmigaBASIC by **Zoltan Szepsi**
Life, Part I: The Beginning by **Gerald Hull**
The ultra-complex nine bit solution to the "Game of Life."
Amiga Virus! by **John Foust**
CLI Arguments in C by **Paul Castonguay**
MIDI Interface Adapter by **Barry Massoni**
Amiga 1000-style MIDI interfaces can fit A2000s or 500s
Modula-2 by **S. Falwiszewski** Part 1: command line calculator
AmigaNotes by **Rick Rae** audio changes made in the A500 &A2000.
Animation for C Rookies: Part III by **M. Swinger** double-buffering.
The Big Picture by **Warren Ring** Assembly language programming
Karate Kid Review by **Stephen R. Pietriowicz**
GO! 64 review by John Foust, James O'Keane, and Rick Wirch
Three C-64 experts investigate a new Amiga 64 emulator.
A-Talk-Plus Review by **Brendan Larson**
Calligrapher Review by **John Foust**
Animator: Apprentice Review by **John Foust**
Playing Dynamic Drums on the Amiga by **David N. Blank**
WordPerfect Review by **Steve Hul**
Insider/Kwikstart Review by **Ernest P. Viveiros Sr**
RAM & ROM expansion: Comments and installation tips.
Forth! by Jon Bryan DumpRPort utility for your Multi-Forth toolbox.
As I See It by **Eddie Churchill** Digi-Paint, Portal, &Videoscape 3D.
The Commodore Show and AmiExpo: New York!
Plus a great collection of monthly columns...

Volume 3 Number 1 1988

AmigaNotes by **Richard Rae** Amiga digital music generation .
C Animation Part IV by Michael Swinger
Forth by John Bryan Sorting out Amiga CHIP and FAST memory
The Big Picture by Warren Ring Daring assembler language programming; CLI system calls and manipulating disk files.
68000 ASsembly Langueage Programming by Chris Martin "Create a multi-color screen without using Intuition routines!"
Modula-2 Programming by S. Falwiszewski A new modula-2!
Amicus Network Special Report: Fall COMDEX by J. Foust
The ultimate Video Accessory: Part II by Larry White
Life: Part II by Gerald Hull The Amiga blitter."
FormatMaster: Professional Disk Formatting Engine byC.Mann Put Batch language to work on the drudgery of disk formatting .
BSpread by Brian Catley full featured AmigaBASIC spreadsheet!
AmigaForum Transcript ed. by Rick Rae Amiga's Dave Haynie.
Haicalc Review by Chuck Raudonis easy to use, spreadsheet.
VIP Professional Review by S. Mitchell Manage stock portfolio
Money Mentor Review by S.Kemp Personal finance system.
Investor's Advantage Review by Richard Knepper
plus "Poor Man's Guide to the Stock Market."
Plus a great collection of monthly columns...

Volume 3 Number 2 1988

Laser Light Shows with the Amiga by Patrick Murphy
Lasers and the Amiga: A Dazzling Tandem
The Ultimate Video Accessory: Partlll by Larry White
Take the final stepstoward designing your own videos.
Our First Desktop Video by Larry White
Step-by -step guide to organizing & presenting your Amiga video.
Hooked on the Amiga with Fred Fish interview by Ed Berkovitz.
Photo Quality Reproduction with the Amiga and Digi-View
by Stephen Lebans
Balancing your Checkbook with WordPerfect Macros by S.Hull
Hand your checkbook worries over to the Amiga.
More Basic Text by Bryan Catley easier text on an Amiga screen
Life: Part III by Gerald Hull
Sries winds up with famed nine-bit calculation & source to LIFER.
Solutions to Linear Algebra through Matrix Computations
by Robert Ellis
Simplify matrix algebra with basic operations & routines.
Modula-2 Programming by Steve Falwiszewski
Catching up with Calc-a source follow-up.
68000 Assembler Language Programming by Chris Martin
Graphics- Part II of Assemgram.
Arazok's Tomb interview by Kenneth E. Schaefer
AIRT by S. Falwiszewski innovative icon-basede program. lang.
Forms In Flight by S. Pietrowicz Render & Animate 3D objects
Silicon Dreams and the Jewel of Darkness by K E. Schaefer
Leisure suit Larry by Kenneth E. Schaefer
Two New Entries From Microbiotics by John Foust
M501 Expansion & Starboard II MultiFunctionboard.
Mindlight 7 and People Meter by John Foust
Phantasie Ken E. Schaefer Amazing Phantasie Character Editor.
Plus a great collection of monthly columns...

Volume 3 Number 3 1988

Desktop Video,Part IV by Larry White
Put all the pieces together-the desktop video commercial.
The Hidden Power of CLI Batch File Processing by J. Rothman
Make your Amiga easier to use with CLI Batch files.
A Conference with Eric Graham edited by John Foust
The mastermind behind Sculpt 3D and Animate 3D.
Perry Kivolowitz Interviewed by Ed Bercovitz Amiga insights from a major developer and personality.
Jean "Moebius" Giraud Interviewed by Edward L. Fadigan
Avant-garde art comes to the Amiga-in dazzling form.
PAL Help by Perry Kivolowitz A1000 expansion reliability.
Boolean Function Minimization by Steven M. Hart
A useful digital design tool in AmigaBASIC.
Amiga Serial Port and Midi Compatibility for Your A2000! by L. Ritter and G. Rentz Add an A1000-style serial port to the A2000!
Electric Network Solutions the Matrix Way by Robert Ellis
Engineers! Practicle routines for using matrix algebra.
The A.M.U.G. BBS List compiled by Joe Rothman, Chet Solace, & Dorothy Dean 514 BBS phone numbers in the U.S. & Canada.
FACC II reviewed by Graham Kinsey Speed your floppy drives.
Uninvited reviewed by K E. Schaefer
Flow reviewed by Pamela Rothman brainstorms into mental art.
Benchmark Modula-2 Complier reviewed by Richie Bielak
Modula-2 Programming by Steve Falwiszewski
The gameport device and simple sprites in action.
AmigaNotes by R. Rae A1000! Software-switchable output filter.
Roomers by The Bandito AmiExpo, Kickstart 1.4, Commodore
The Big Picture by Warren Ring-- Unified Field Theory!!
Plus a great collection of monthly columns...

Volume 3 Number 4 1988

Highlights from AmiExpo, Los Angeles by Steve Hull
Writing a SoundScape Patch Librarian T. Fay System Exclusive
Upgrade Your A1000 to A500/2000 Audio Power—by H.Bassen
Modifications to help your A1000 make sweet music, too!
Amiga Audio Guide Listing of all Amiga audio products.
Gels In Multi-Forth by John Bushakra
Macrobotics by Patrick J. Horgan Ease the trauma of assembly language programming.
Amiga Audio Sources The folks behind all those audio products.
Take Five! by Steve Hull five Amiga games reviewed.
Amiga Notes by Rick Rae A basic tour of Amiga audio.
The Ultimate Video Accesory, Part V by Larry White
Bug Bytes by John Steiner
The Big Picture by Warren Ring Part II Unified Field Theory.
Roomers by The Bandito
Hardware hijinx ... Toasted video ... the dream Amiga ... and more!
In the Public Domain by C.W. Flatte
Time Bandit review by Keith Conforti
AudioMaster review by B. Larson Real-time digitizing samples.
Music Mouse review by J Henry Lowengard
Making music without lifting a finger from the mouse.
Amiga-Tax/Canadian Version review by Ed Bercovitz
A Canadian income tax planning, preparation, & analysis package.
SAM BASIC review by Bryan Catley
A new BASIC which exploits even more unique Amiga features.

Volume 3 Number 5 1988

Interactive Startup Sequence by Udo Pernisz
The Command Line part 1by Rich Falconburg
AmigaTrix III by Warren Block—Tips and tidbits to ease Amiga life
Amiga Product Guide: Hardware Edition
Proletariat Programming by P.Quaid—Public domain compilers
The Companion by P.Gosselin Amiga's Event Handling capability.
MindLight 7 reviewed by David N. Blank
VideoScape 3-D 2.0 reviewed by David Hopkins
Extend reviewed by Bryan D. Catley—An AmigaBASIC extension
AssemPro reviewed by Stephen Kemp
Opening a door to assembly language programming.
APL.68000 reviewed by Roger Nelson
Book Reviews by Richard Grace—Three "C" programming texts.
CBTREE reviewed by Michael Listman
A tidy collection of functions to aid the C programmer.
The Big Picture by Warren Ring
The three-part Unified Field Theory winds up
Modula-2 by Steve Falwiszewski
Termination modules for Benchmark and TDI compilers.
68000 Assembly Language by Chris Martin
Peeling away the complication of display routines.
Plus a great collection of monthly columns...

Volume 3 Number 6 1988

Bear Time Reviewed by Steve Carter
What makes this inexpensive A1000 battery-backed clock tick?
Acquisition Reviewed by David N. Blank
A look inside the latest release of a powerful relational database.
Butcher 2.0 Reviewed by Gerald Hull
A tidy collection of diverse image processing utilities.
Reassigning Workbench Disks by John Kennan
Endless disk swapping comes to a merciful end.
Product Guide: Software Tools Edition
A listing of all the products you need to put your Amiga to work.
An IFF Reader in Multi-Forth by Warren Block
Create an easy to use IFF reader in Multi-Forth.
Basic Directory Service Program by Bryan Catley
A programming alternative to the GimmeeZeroZero windows.
C Notes from the C Group by Stephen Kemp
A beginner's guide to the power of C programming.
An Amiga Forum Conference with Jim Mackraz
The Amiga market as seen by the "Stepfather of Intuition."
Son of Seven Assemblers Reviewed by Gerald Hull
A ocmparative battle between seven native-code assemblers.
The 1988 Commodore Amiga Developers Conference
A look inside the conferences held in Washington, D.C.
Amiga Working Groups by Perry Kivolowitz and Eric Lavitsky
An outline of the innovative Amiga Working Groups concept.
The Command Line by Rich Falconburg
Exploring the multi-talented LIST command.
Plus a great collection of monthly columns...

Volume 3 Number 7 1988

Look, Up On the Screen, It's an Ami... It's a Pro... It's SuperGen reviewed by Larry White—Genlock comparisons
An Interview with "Anim Man," Gary Bonham by B. Larson
An animated conversation with the man behind the format.
The Amiga at Spring COMDEX in Atlanta by Ed Bercovitz
Amiga Product Guide: Video/Graphics Edition
Thirteen pages devoted to the Amiga's dazzling strong suit.
The Developing Amiga by Steve Pietrowicz
Developers' notes: PD vs. shareware vs. freely distributableetc.
Roll Those Presses! by Barney Schwartz
Welcome to the dandy, demanding world of desktop publishing!
Linked Lists in C by W. E. Gammill Put dynamic memory to work!
FrameGrabber Preview by Oran Sands
Capturing an image can now be as fast as punching a single key!
A First Look at Interchange reviewed by David Hopkins
Bridge the gap between those incompatible animation packages.
Perfect Vision reviewed by Bryan Catley
Capture, digitize and save pictures from any video source.
ProWrite 2.0 Review reviewed by Pamela Rothman
A graphic word processor specializing in efficient editing.
Doug's Math Aquarium: The Art of Mathematics by R. Bielak
Bear Products MegaRex II Expansion RAM by Steve Carte
The Command Line by Rich Falconburg
Amiga Notes by Rick Rae
The Other Guys' Synthia: A digital synthesizer features won't stop.
C Notes from the C Group by Stephen Kemp
Weathering the unknown "C" of basic object and data types.
Plus a great collection of monthly columns...

Volume 3 Number 8 1988

The Command Line by Rich Falconburg
The journey into the CLI continues.
The Developing Amiga by Stephen R. Pietrowicz
A gaggle of great programming tools.
Modula-2 Programming by Steve Falwiszewski
Libraries and the FFP and IEE Math Routines.
C Notes from the C Group by Stephen Kemp
Arrays and pointers unmasked.
Dark Castle reviewed by Keith Conforti—The Black Knight lurks
Ports of Call reviewed by Julie Landry
Leatherneck reviewed by Michael Creeden-Rambo's not so tough!
Capone reviewed by Joyce and Robby Hicks- Light Guns blaze
Casino Fever reviewed by Michael T. Cabral—Vegas on Amiga
Ferrari reviewed by Jeffery Scott Hall—Start your engine
Arkanoid reviewed by Graham Kinsey—"blockbuster"
Ebonstar by Keith Conforti —biack hole treking.
Deluxe Productionsreviewed by Harv Laser—Video wizardry
Game Pizazz by Jeffery Scott Hall— Register your questions here.
TrackMouse by Darryl Joyce
Convert a standard Atari trackball into a peppy Amiga TrackMouse.
Amiga Interface for Blind Users reviewed by Carl W. Mann
An ingenious interface that opens the Amiga to even more users!
Video In the Sunshine State reviewed by Stephen R. Pietrowicz
RGB Video Creations hosts a video unveiling!
Amiga Product Guide: Games Edition
Tumblin' Tots by David Ashley—assemply language program.
Plus a great collection of monthly columns...

Volume 3 Number 9 1988

The Kldeo Tapes by John Dandurand
A Georgia elementary school puts desktop video to work.
Speeding Up Your System by Tony Preston floppy disk caching
Amiga Product Guide: Education Edition
Everything you need to send your Amiga to the head of the class.
Computer Aided Instruction by Paul Castonguay
Authoring system in AmigaBASIC.
Gels In Multi-Forth, Part II: Screenplay by John Bushakra
Make the IFF converter from Part I easy to use-gadgets, menus,etc.
AmiExpo Midwest '88 by Michael T. Cabral
After taking the coasts by storm, the Amiga wows Chicago
Intellitype by Harv Laser—Learning to type made easy...and fun?
Shakespeare by Barney Schwartz—Desktop publishing in full color.
XSpecs 3D by Steve Hull—A new dimension in Amiga graphics.
AmigaNotes by Richard Rae—How IFF sound samples are stored?
Take Five! by Steve Hull—Beat the back-to-school blues !
The Command Line by Rich Falconburg—continuing tour of CLI.
Hot on the Shelves by Michael T. Cabral & Michael Creeden
What do you get when you combine intense war strategy with a monochrome monitor and desktop presentation? Check it out.
Bug Bytes by John Steiner
C Notes from the C Group by Stephen Kemp
Operators, expressions, and statements in C uncovered.
Roomers by The Bandito Can Apple IIgs Plus keep Amiga away?

Volume 3 Number 10 1988

A First Look At Deluxe PhotoLab reviewed by David Duberman
A paint package, poster-maker, and image processing program
DiskMaster reviewed by Steve Hull —file management utility.
DSM: A MC680000 Disassembler reviewed by Gerald Hull
Looking for easily modifiable, assembler-ready code?
FBasic Language System reviewed by Patrick Quaid
BASIC compiler and development system.
Hot on the Shelves by Michael T. Cabral—Deviant dice, gripping gray scales, color cartography, mauling modems, and much more.
The Command Line by Rich Falconburg
NEWCLI: A painless way to create a new console window.
The Developing Amiga by S. Pietrowicz Usenet—24-Hour News
C Notes from the C Group by Stephen Kemp —loops
Roomers by The Bandito WP wars, ignominious interfaces, & more
PD Serendipity by C.W. Flatte—Fred Fish collectionpasses 150.
Comparison of MultiScan Monitors by Steven Bender
Five multiscan alternatives square off on the desktop.
Record Keeping for Free-lancers:
A Superbase Professional Tutorial by Marion Deland
Record keeping system for free-lance photographers and others.
On The Crafting of Programs by David J. Hankins—A look at optimization kicks off a series of articles on programming savvy.
Bob and Ray Meet Frankenstein by Robert D'Asto—Create, animate, and metamorphose graphics objects in AmigaBASIC.
Digital Signal Processing in AmigaBASIC by Robert Ellis
Perform your own digital experiments with Fast Fourier Transforms.
HAM & AmigaBASIC by Bryan Catley—Pack your AmigaBASIC programs with many of the Amiga's 4096 shades!
CAI—Computer Aided Instruction: Part II by Paul Castonguay
The Editor program wraps up our authoring system in AmigaBASIC.

Volume 3 Number 11 1988

Desktop Publishing with Professional Page by Barney Schwartz
tutorial in document creation, plus some jazzy enhancements.
Game Pizzazz by J. Hall gaming hints, tips, high-score secrets.
Structures In C by Paul Castonguay C programming in an nutshell.
On The Crafting of Programs by D. Hankins speed up your progs.
Desktop Video VI: Adding the Third Dimension by Larry White
Unraveling the complexity of 3D for your video creations.
A2000 Hard Drive Round Up by Sheldon Leemon
Keyclick by Mike M. Duppong a typewriter click in your keyboard.
More Linked Lists in C: Techniques and Applications by Forest W. Arnold Procedures for managing lists, storing diverse data types in the same list, and putting lists to work in your programs.
BASIC Linker by Brian Zupke Combine individual routines from your program library to create an executable program.
The Developing Amiga by Steven Pietrowicz
A look at mysteries and successes behind efficient beta testing.
Modeler 3D Preview reviewed by David Hopkins
A peek inside a new, open-ended 3D package.
APro Draw Graphics Tablet reviewed by Keith Conforti
Artists! Meet the future of Amiga graphics.
StarGlider II reviewed by Jeffery Scott Hall
Those irritating Ergons are back for another laser-lashing.
WShell reviewed by Lawrence Lichtman CLI substitute,
Hot on the Shelves by M. Cabral viruses, music, microfiche mastery
PD Serendipity by C.W. Flatte Fred Fish disks 149-152.
Roomers by The Bandito Golden RAM, 16-bit videogames, CD-I, another HAM skirmish ... what could possibly be NeXT?

To be continued.....

example, the previous paragraph mentioned two offsets of 86 and 15 bytes to obtain the port address and the signal number. These numbers can be replaced by the labels wd_UserPort and MP_SIGBIT respectively. Adding the statement, INCLUDE "intuition/intuition.i" will get the two values from this include file. This is done so that if the structure is changed in future revisions of the operating system, the program will only need to be re-assembled with new updated include files. Personally, I loathe include files, and prefer to add my own EQU directives to resolve the symbolic labels, but this would have made the listing too long here.

My biggest complaint about using include files is that they "hide" important information from you. If I had used symbolic labels for accessing the fields of the IntuiMessage, I might not have noticed that all of the fields are adjacent in memory, and that post-increment addressing mode could be used. The 2 fastest ways to access blocks of memory is via post-increment and pre-decrement modes. I use them wherever possible though this may make it more difficult to update the program if the structures are changed in new operating software. If you are interested in churning out code, and execution speed is a minor consideration, program in C. By the way, extreme use of indexed addressing mode is one reason why Amiga libraries and devices are rather slow.

Once all the info from the message is stored somewhere (in this case 68000 registers), it is imperative that the message be replied. Intuition will free the memory. We have all the info we want. We don't need the message anymore.

The class of the message is determined at E9. Remember that each of the available classes has its own bit. Depending upon which bit is set, that is the class. Refer to the above IDCMPflags chart for the bit assignment. At this point, the program branches to the proper handler for the class. For brevity, I have substituted dummy handlers for all of the classes except MenuPick (not one of my attributes). In future articles, some useful handlers for the other classes will be devised and substituted for the dummy ones.

If the message is MENUPICK, then _main branches to decode_menu. The important field is the Code, which was stored in d5. For a MENUPICK, the Code contains the selected subitem number (if there was one), the item number, and the Menu number where the item resides. All these numbers are bit packed into that one WORD in d5. First, we check if the user backed off without selecting anything at all. If so, d5 will = -1 (MENUNULL). If something was selected, the menu number must be determined since the program has 3 menus. By AND'ing with $1F, the bits that represent the menu number are isolated from the bits representing the item number and the subitem number. Once the menu is determined, the appropriate handler is called. The item number within this menu must be determined next. This is done precisely the same way as determining the menu number, except that the bits which need to be examined must be shifted to the low bits of the register. Finally, if a subitem of an item was selected, this needs to be determined.

(continued)

Let's assume that the user selected the Pen3 subitem in the Project menu. The class of the IntuiMessage is MENUPICK, so _main branches to decode_menu. At F1, the menu number bits within the Code are isolated and evaluated. Since the Project Menu is the first in the linked list of menus, it is menu #0. The branch is taken to F4 which calls do_menu0. At G1, the item number bits within the Code are isolated and since Pen3 is a subitem of Color (which is the Item #0 in the Project menu) the branch is taken to G4. The routine, do_colors, isolates which of the 3 subitems under Color was selected, and sets the foreground pen color to that color register + 1. The reason 1 is added is so that the foreground pen can't be set to register 0, which is the background color.

The other two menus simply print one of 3 messages to the window, and erase the same messages (by printing spaces over the area). The byte PrintMode determines which operation. Rather stupid, right? This program is meant to be only a skeleton example of using Intuition. Experiment with adding menus and items, and more meaningful handlers.

When returning back to _main from any one of the handlers, the code always branches back to E4 to check for any more messages at the port. GetMsg only returns one message at a time. It is entirely possible that Intuition has "queued" several messages. GetMsg returns them in the order that they arrived at the port. When the last message is removed from the port and processed, the program has nothing more to do, and so tells Exec to "put it to sleep" with a call to Wait. The signal mask for the UserPort is passed to this function in d0. In essence, the program is asking Exec to allow resumption only when Intuition puts a message to the port and signals the task. Never let a task go to sleep before all messages have been gotten and replied.

The routine, print_message, uses a structure of my own creation. Each string has a MsgStruct. It contains the address of the NULL-terminated string (LONG) and the X and Y window co-ordinates where the string will be printed (WORDS). Since I need to output one of 5 possible strings, there are 5 of these structures all adjacent in memory beginning at the label MsgStruct. I prefer to have related structures adjacent in memory rather than using the popular linked list approach. This preference is based on execution speed. With adjacent structures, post-increment and pre-decrement addressing modes can be used. Also, I only need to know the base of the first structure, and the size of one structure in order to locate the desired structure. Searching linked lists can be very time-consuming. Also, I try to make the size of a structure a power of 2, (i.e. 8, 16, 32, etc). This is so that instead of calling a multiplication routine to find the offset from structure #1, I can use left shifts or simple adds (as demonstrated at line K2). Arrange the members of the structure in the order that they need to be accessed so that post-increment addressing mode can be used (as in K3).

The last aspect of the program is exiting it. The user exits by either clicking on the close gadget (which sends a CLOSEWINDOW message to the port) or by selecting the Quit item from the Project menu. Both actions clear the byte Quit which was initially set to a 1. This happens in line E16 of _main, and line G2 of do_menu0, respectively. Note that I don't immediately branch to an exit routine. This is because there might be more IntuiMessages queued at the port. Never exit a program before all messages sent to the task are replied, so that these messages can be freed. The point where Quit is examined to see if the user wishes to exit, is when GetMsg returns d0 = 0 (indicating no more messages). This occurs in line E5 of _main. If so, then the branch to E15 is taken, and quit_all closes/deAllocates everything that was opened/Allocated in exactly the opposite order.

By using this example program as a starting point, you should be able to quickly develop a program in assembly language that utilizes the most unique aspects of the Amiga: Intuition and libraries. Later, I will present some examples of how rawkey and mouse messages might be interpreted, as well as adding gadgets to the window. These will be "plug-in" modules to replace the dummy calls, so for now, assemble the program and experiment with menus, fonts, and outputting text.

FINAL NOTES

If you are using the Manx Asm, disable the small code, small data feature.

Execution speed is best obtained by the following:

1) Using all 14 of the 68000's registers.
2) Using post-increment and pre-decrement addressing modes instead of indexed.
3) Passing values in registers wherever possible.
4) Using conditional branching for loops wherever possible.

The Manx method of speeding up code requires the use of register a4, and may impose limits upon the size of the program. It is best to use the more conventional methods mentioned above.

If you want to eliminate all external references, the _LVO labels can be replaced by the hex or negative decimal absolute values in the chart starting on page D-5 of the RKM Exec manual. Since the libraries and devices are what really undergo changes between system software updates, this is not recommended though.

If you do not have a start-up code and are too lazy to type in the example in the RKM Libraries and Devices (pg. 489), this program can be made to run from the CLI by adding the following code at the beginning of open_libs:

```
   movea.l  4,a6           ;The base address of Exec library is ALWAYS
   move.l   a6,_SysBase    ;stored at address $00000004.
;====OPEN THE DOS LIBRARY======
   moveq    #29,d0
   lea      DOSName,a1
   jsr      _LVOOpenLibrary(a6)
   move.l   d0,_DOSBase
   beq.s    B10

 and in the data section add:

_SysBase dc.l 0
_DOSBase dc.l 0
DOSName  dc.b 'dos.library',0

 Delete the XREF to _SysBase and _DOSBase.
```

Listing One
Example1Code

```
   SECTION Example1Code,CODE

 XREF  _SysBase,_DOSBase
 XREF  _LVOOpenLibrary,_LVOCloseLibrary
 XREF  _LVOCloseFont,_LVOOpenFont,_LVOSetFont
 XREF  _LVOCloseWindow,_LVOOpenWindow
 XREF  _LVOSetMenuStrip,_LVOClearMenuStrip
 XREF  _LVOGetMsg,_LVOReplyMsg,_LVOWait
 XREF  _LVOMove,_LVOText,_LVODraw
 XREF  _LVOSetAPen,_LVOSetBPen,_LVOSetDrMd
 XREF  _LVODelay
 XREF  _LVOFindTask

;++++++++++++++++++++++++++++++++++++++++++++++
; This is the main module where program execution begins.

                    PUBLIC _main   ;needed for Manx Asm
_main:
E1 movem.l  d2-d7/a2-a3,-(a7) ;Save all these regs before we use them.
;===============Open  Everything==============
E2 bsr.s    open_libs         ;Use bsr.s instead of jsr because open_libs
                              ;is close enough to branch there.
   move.b   d0,d1             ;Test for an error in open_libs.
   beq.s    E15               ;If an error, branch to exit the program.
;====Create a mask of the Window's mp_Sigbit=====
E3 movea.l  MyWindow,a0       ;the address of our main window.
   movea.l  86(a0),a3         ;get the UserPort of MyWindow.
   move.b   15(a3),d0         ;get the UserPort's mp_Sigbit number.
   moveq    #0,d7
   Bset.l   d0,d7             ;set this bit # in d7. d7 is now our mask.
;=====Get the message that arrived at our UserPort====
E4 movea.l  a3,a0             ;the address of the UserPort of MyWindow
   movea.l  _SysBase,a6
   jsr      _LVOGetMsg(a6)
   move.l   d0,d1             ;the address of the message.
   bne.s    E7                ;Branch if a message at the port.
;===Check if we are ready to exit the program=====
E5 move.b   Quit,d0           ;If Quit = 0, then we want to exit.
   beq.s    E15
;===Wait for a message to our Window (from Intuition)===
E6 move.l   d7,d0             ;This is the mask of the bits to wait for.
                              ;_SysBase already in a6.
   jsr      _LVOWait(a6)
   bra.s    E4
;====Copy all the info we want from the IntuiMessage====
E7 movea.l  d0,a1
   lea      20(a1),a0       ;get the address of the first field to copy.
   move.l   (a0)+,d6           ;Copy the Class field to d6
   move.w   (a0)+,d5           ;Copy the Code field to d5
   move.w   (a0)+,d4           ;Copy the qualifier field to d4
   movea.l  (a0)+,a2           ;Copy the IAddress field to a2
   move.w   (a0)+,d3           ;Copy MouseX position to d3
   move.w   (a0)+,d2           ;Copy MouseY position to d2
;====Now reply to the message so Intuition can dispose of it
E8                            ;Address of the message is in a1.
                              ;_SysBase is still in a6.
   jsr      _LVOReplyMsg(a6)
;======Decode the Class of the Message=========
E9 Bclr.l   #9,d6
   bne.s    E16               ;Branch if CLOSEWINDOW ($200)
   Bclr.l   #10,d6
   bne.s    E10               ;Branch if RAWKEY ($400)
   Bclr.l   #8,d6
   bne.s    E12               ;Branch if MENUPICK ($100)
   Bclr.l   #5,d6
   bne.s    E11               ;Branch if GADGETDOWN ($20)
   Bclr.l   #4,d6
   bne.s    E14               ;Branch if MOUSEMOVE ($10)
   Bclr.l   #3,d6             ;Branch back if not MOUSEBUTTONS (8)
   beq.s    E4                ;Check for any more messages at UserPort.
;=============RIGHT  MOUSEBUTTON==============
;The qualifier (in d4) tells us whether the user has pressed the button
;down, or is letting it up. In either case, Intuition sends a message.
;Also, we may want to know where the user clicked in the window. The
;MouseX (in d3) and MouseY (in d2) give the co-ordinates.
E13 bsr.s   button
    bra.s   E4
;================DECODE  RAWKEY=====================
;The code (in d5) contains the key # that was depressed. See page 341
;of the RKM Libraries and Devices for the key number map. The qualifier
;(in d4) gives info about whether a ctrl, shift, alt, Amiga key, etc was
;being pressed simultaneously. See page D-116 for the bit masks.
E10 bsr.s   rawkey
    bra.s   E4
;=========SERVICE A GADGET DOWN REQUEST============
;The address of the selected gadget structure is IAddress (in a2).
E11 bsr.s   gadget_down
    bra.s   E4
```

(continued)

```
;================MENU  DECODING=====================
;The code (in d5) gives info about the menu, item, and subitem numbers.
E12 bsr.s    decode_menu
    bra.s    E4
;==================MOUSE  MOVE=====================
;The MouseX (in d3) and MouseY (in d2) give our co-ords in the window.
E14 bsr.s    mouse_move
    bra.s    E4
;==================CLOSE  WINDOW  GADGET=============
E16 clr.b    Quit                ;Indicate that user wants to exit.
    bra.s    E4                  ;Answer any remaining intuimessages.
;==================EXIT  PROGRAM=====================
;Here's how we get out of the program. Quit must have been set to 0.
E15 bsr.s    quit_all
    movem.l  (a7)+,d2-d7/a2-a3   ;restore the original contents
    moveq    #0,d0               ;return error code
    rts

;++++++++++++++++++++++++++++++++++++++
;  Open intuition, graphics libs, font, window, set menu
;  returns d0 = 1 if successful, 0 if something failed.

open_libs:
;======Open The Intuition Library=======
B1  moveq    #29,d0              ;the version # that we want opened.
    lea      IntuitionName,a1    ;address of the string
'intuition.library',0
    movea.l  _SysBase,a6         ;OpenLibrary is in the Exec Library.
    jsr      _LVOOpenLibrary(a6)
    move.l   d0,_IntuitionBase   ;the base address of the Intuition lib.
    beq.s    B10                 ;If 0, there was an error. Exit with d0
                                 ;equals zero, so that _main will know.
;======Open The Graphics Library========
B2  moveq    #29,d0              ;the version # that we want opened.
    lea      GfxName,a1          ;address of the string
'graphics.library',0
                                 ;_SysBase is still in register a6.
    jsr      _LVOOpenLibrary(a6)
    move.l   d0,_GfxBase
    beq.s    B10                 ;If 0, then there was an error. Exit.
;========Find this Task's address========
B3                               ;_SysBase is still in a6.
    suba.l   a1,a1               ;passing a 0 means "find this task"
    jsr      _LVOFindTask(a6)
    move.l   d0,MainTaskAddr     ;the address of our _main Task
;========Open the Topaz 8 Font==========
B4  lea      TextAttr,a0         ;address of Text Attribute structure.
    movea.l  _GfxBase,a6         ;OpenFont is in the Graphics library.
    jsr      _LVOOpenFont(a6)
    move.l   d0,FontPtr
    beq.s    B10                 ;If 0, the font couldn't be opened.
;=========Open the main window==========
B5  lea      WindowStruct,a0     ;the address of our window structure.
    movea.l  _IntuitionBase,a6   ;OpenWindow is in Intuition lib.
    jsr      _LVOOpenWindow(a6)
    move.l   d0,MyWindow
    beq.s    B10                 ;If 0, our window couldn't be opened.
Exit.
;====Get Pointer to Window's RastPort=====
B6  movea.l  d0,a0
    movea.l  50(a0),a1
    move.l   a1,RastPort         ;the address our this window's
rastport.
;====Set the Font for this window to Topaz 8====
B7                               ;rastport address is in a1.
    movea.l  FontPtr,a0          ;the address of the opened Topaz Font.
    movea.l  _GfxBase,a6         ;SetFont is in the Graphics lib.
    jsr      _LVOSetFont(a6)
;====Attach our menus to the window======
B8  lea      ProjectMenu,a1      ;The address of the first menu in list.
    movea.l  MyWindow,a0         ;The address of window to attach to.
    movea.l  _IntuitionBase,a6   ;SetMenuStrip is in Intuition lib.
    jsr      _LVOSetMenuStrip(a6)
;====Set the Draw Mode, Back and ForeGround Pens====
B9  moveq    #1,d0               ;mode = JAM2
    movea.l  RastPort,a1
    movea.l  _GfxBase,a6
    jsr      _LVOSetDrMd(a6)
    moveq    #1,d0               ;front pen = color reg 1
    movea.l  RastPort,a1
    jsr      _LVOSetAPen(a6)
    moveq    #0,d0               ;back pen = color reg 2
    movea.l  RastPort,a1
    jsr      _LVOSetBPen(a6)
;====Indicate that everything worked=====
    moveq    #1,d0               ;If we got here, show success by d0= 1.
B10 rts

;++++++++++++++++++++++++++++++++++++++
; Closes window, font, graphics, intuition libs

quit_all:
;========Close the Topaz 8 Font=========
C1 move.l    FontPtr,d0          ;Check if the Font was ever opened.
   beq.s     C2                  ;Don't try to close it if not opened!!
   movea.l   d0,a1
   movea.l   _GfxBase,a6         ;CloseFont is in the Graphics lib
   jsr       _LVOCloseFont(a6)
;===========Close the Window============
C2 move.l    MyWindow,d0
   beq.s     C3
   movea.l   d0,a0
   movea.l   _IntuitionBase,a6   ;CloseWindow is in the Intuition lib.
   jsr       _LVOCloseWindow(a6)
;=====Close Whichever Libs are Open=====
C3 movea.l   _SysBase,a6         ;CloseLibrary is in Exec Library.
   move.l    _GfxBase,d0
   beq.s     C4
   movea.l   d0,a1
   jsr       _LVOCloseLibrary(a6)
C4 move.l    _IntuitionBase,d0
   beq.s     C5
   movea.l   d0,a1
                                 ;_SysBase is already in a6.
   jsr       _LVOCloseLibrary(a6)
C5 rts

;++++++++++++++++++++++++++++++++++++++++++++++
; For now, we just recognize the following intuimessages, but we only
; call a dummy routine. Later, a useful handler can be made for each.
button:
rawkey:
gadget_down:
mouse_move:  rts


;++++++++++++++++++++++++++++++++++++++++++++++
;This decodes which menu # has been selected.
;The Code is passed in d5 (WORD).

decode_menu:
;====Determine which of the 3 Menus (Project, Print, or Erase)===
F1 move.w   d5,d0            ;Duplicate the Code. (We need it later).
   cmpi.w   #-1,d0
   beq.s    F5               ;Skip if MENUNULL (-1), i.e. nothing selected
   andi.w   #$1f,d0          ;Isolate the menu # from the item and subitem.
   beq.s    F4               ;Branch if Menu #0 (Project)
   subq.w   #1,d0
   beq.s    F3               ;Branch if Menu #1 (Print).
                             ;Otherwise, must be Menu #2 (Erase).
;======Decode the Erase Menu=======
F2 bsr.s    do_menu2
   rts
;=====Decode the Print Menu========
F3 bsr.s    do_menu1
   rts
;=====Decode the Project Menu======
F4 bsr.s    do_menu0
F5 rts

;++++++++++IMPLEMENT MENU #0 (Project Menu)++++++++++
; The Code has been passed in d5 (WORD).

do_menu0:
;=======Determine which item========
G1 move.w   d5,d0            ;Duplicate Code. We still may need it later.
   lsr.w    #5,d0            ;Shift the item # bits into lowest bits of reg.
   andi.w   #$3f,d0          ;Isolate the Item # from the Menu & subitem #.
   beq.s    G4               ;branch if Item #0 (Color)
   subq.w   #1,d0
   beq.s    G3               ;branch if Item #1 (Info)
   subq.w   #1,d0
   bne.s    G5               ;branch if not Item #2 (Quit)
;====Indicate to Main that we wish to exit the program====
G2 clr.b    Quit
   rts
;===========Do the Info Item===========
G3 bsr.s    what_is_this_crap
G5 rts
;=======Do the color Item's SubItem=====
G4 bsr.s    do_colors
   rts
;++++++++++IMPLEMENT MENU #1 (Print Menu)++++++++++
do_menu1:
;=======Determine which item=======
H1 lsr.w    #5,d5            ;shift Item number bits
   andi.l   #$3f,d5          ;Isolate Item # and pass it to print_message.
   Bset.b   #0,PrintMode     ;PrintMode = 1 (Print)
   bsr.s    print_message
   rts
;++++++++++IMPLEMENT MENU #2 (Erase Menu)++++++++++
do_menu2:
;=======Determine which item=======
H2 lsr.w    #5,d5
   andi.l   #$3f,d5
   Bclr.b   #0,PrintMode     ;PrintMode = 0 (Erase)
   bsr.s    print_message
   rts
;++++++++++++++++++++++++++++++++++++++++++++++
; Determines which of the 3 drawing pen colors we have chosen.

do_colors:
;=======Get the SubMenu Selection (Pen color)======
```

```
J1 moveq   #11,d0
   lsr.w   d0,d5            ;Shift our copy of code to the subitem field.
   andi.b  #$1f,d5          ;Isolate the subItem # from item and menu #.
;======Set the foreground pen to this color # + 1====
   moveq   #0,d0
   move.b  d5,d0            ;color register # must be in d0.
   addq.b  #1,d0            ;don't allow same as the background color
   movea.l _GfxBase,a6
   movea.l RastPort,a1
   jsr     _LVOSetAPen(a6)
   rts
;++++++++++++++++++++++++++++++++++++++++++++++++
; Print a message to the window. Passed the message # in d5.

print_message:
   movem.l a2/a3,-(a7)      ;save these registers before we use them.
K1 movea.l RastPort,a2
   movea.l _GfxBase,a6
;====Get the address of the message======
K2 lsl.w   #3,d5
   movea.l d5,a0            ;the message (item #) * 8
   adda.l  #MsgStruct,a0    ;add to the base of the first structure
   movea.l (a0)+,a3         ;get the address of the string to print
;===Get the XY positions where the string to be displayed====
K3 moveq   #0,d0
   move.w  (a0)+,d0         ;the x co-ordinate
   moveq   #0,d1
   move.w  (a0),d1          ;the y co-ordinate
;====Move to Position where message to be Output====
K4 movea.l a2,a1            ;our window's RastPort
                            ;_GfxBase already in a6.
   jsr     _LVOMove(a6)
;====Determine the # of Bytes in the String======
K5 movea.l a3,a0            ;Duplicate the starting address of the
string
K6 move.b  (a3)+,d0         ;Is this the terminating NULL byte?
   bne.s   K6               ;Branch if not, to check the next byte.
   subq.l  #1,a3            ;Backup to the NULL byte.
   suba.l  a0,a3            ;determine the number of bytes from head to
tail
;====Check whether to print the message, or spaces====
K7 move.b  PrintMode,d1
   bne.s   K8               ;Branch if "Print"
   lea     Spaces,a0        ;Erase by outputting spaces over old text.
;======Print out the message======
K8 move.l  a3,d0            ;# of bytes to output.
                            ;The address of the string is in a0.
                            ;_GfxBase is still in a6.
   movea.l a2,a1            ;RastPort
   jsr     _LVOText(a6)
   movem.l (a7)+,a2/a3
   rts
;++++++++++++++++++++++++++++++++++++++++++++++++
; Flashes an explanation to the window (prints, delays for a interval,
; and then erases the message). D5 can be freely used, since _main saved
; it for us, and doesn't expect it back intact.

what_is_this_crap:
    move.l  d2,-(a7)            ;save d2 before using
    moveq   #3-1,d2             ;flash it 3 times
;==========Print out the 2 line message========
L1  Bset.b  #0,PrintMode
    moveq   #3,d5               ;print message #3
    bsr.s   print_message
L2  moveq   #4,d5               ;print message #4
    bsr.s   print_message
;==========Delay for an interval==============
L3  moveq   #90,d1
    movea.l _DOSBase,a6
    jsr     _LVODelay(a6)
;==========Erase the 2 line message============
    Bclr.b  #0,PrintMode        ;PrintMode = "Erase"
L4  moveq   #3,d5               ;erase message #3
    bsr.s   print_message
L5  moveq   #4,d5
    bsr.s   print_message
;==========Delay for an interval==============
L6  moveq   #50,d1
    movea.l _DOSBase,a6
    jsr     _LVODelay(a6)
    Dbra    d2,L1               ;flash it again
    move.l  (a7)+,d2
    rts
  SECTION ExamplelData,DATA
WindowStruct:
   dc.w   0      ;leftedge
   dc.w   0      ;topedge
   dc.w   640    ;width
   dc.w   200    ;height
   dc.b   0      ;detailpen
   dc.b   1      ;blockpen
   dc.l $778     ;IDCMPflags =
  ;RAWKEY|MENUPICK|GADGETDOWN|GADGETUP|MOUSEMOVE|MOUSEBUTTONS|CLOSEWINDOW
   dc.l $120F    ;Window flags =
  ;ACTIVATE|WINDOWDEPTH|REPORTMOUSE|SMARTREFRESH|WINDOWSIZING|WINDOWDRAG
   dc.l   0      ;FirstGadget (for now, no gadgets)
   dc.l   0      ;Checkmark
```



```
   dc.l   Title  ;ptr to title
   dc.l   0      ;ptr to screen
   dc.l   0      ;ptr to bitmap
   dc.w  25      ;minWidth
   dc.w  20      ;minHeight
   dc.w 640      ;maxWidth
   dc.w 200      ;maxHeight
   dc.w   1      ;WB screen type

;==========THE PROJECT MENU===========
 ;This is a Menu structure.
ProjectMenu:
   dc.l PrintMenu      ;The address of the next Menu
   dc.w 0,0            ;X, Y offset from upper left corner of window
   dc.w 90,0           ;the width and height of the select box
   dc.w 1              ;menu Flags = MENUENABLED
   dc.l ProjectTitle   ;The ascii string name of Menu #0
   dc.l ColorItem      ;The address of the first MenuItem structure
   dc.w 0,0,0,0        ;these are for Intuition's use only

;======The Items in Menu0========
 ;These are MenuItem structures for the preceding Menu Structure.
ColorItem dc.l InfoItem      ;the address of the next Item
          dc.w 0,0           ;X,Y from top left of menu box
          dc.w 100,10        ;width, height of the select box
          dc.w $52           ;item flags
          dc.l 0             ;Mutual Exclude
          dc.l ColorText     ;address of the IntuiText for this Item
          dc.l 0             ;SelectFill, for an alternate image
          dc.b 0             ;the ascii value for the keyboard shortcut
          dc.b 0             ;pad
          dc.l Pen1SubItem   ;address of Color's first SubItem
          dc.w 0           ;NextSelect, for drag select (Intuition's use)
InfoItem  dc.l QuitItem
          dc.w 0,10,100,10
          dc.w $96     ;item flags = COMMSEQ|ITEMTEXT|HIGHBOX|ITEMENABLED
          dc.l 0,InfoText,0
          dc.b 'I'           ;Command, Keyboard shortcut of 'I'
          dc.b 0
          dc.l 0
          dc.w 0
QuitItem  dc.l 0             ;No more items in Project Menu
          dc.w 0,20,100,10,$56
          dc.l 0,QuitText,0
          dc.b 'Q',0
          dc.l 0
```

(continued)

Amazing COMPUTING™

Reader Friendly - Reader Service

Wow, AC has done it again!
Due to the tremendous response of our original Reader Service Program, Amazing Computing announces an easier way to contact AC advertisers: The AC Reader Service Card Mailer.

To use the AC system, simply locate all the Amazing Advertisers you wish to contact below, find their appropriate AC Reader Service Card Numbers, and mark them on the card to the right. Fill in your address and mail the card. It is that easy!

The AC Difference
OK, this is not really a new idea. But, we have added a new twist. Gone are the awkward lines "Reader Service Card Number XXX" that reside somewhere on an advertisement. Instead, AC has listed all the advertisers in the Index Of Advertisers with their page numbers and special codes.

Now it is easy to find all the information you want for any AC advertiser in just one place. As always, Amazing Computing has placed their readers first!

Index of Advertisers

```
          dc.w 0
;====The Pen subitems for the Color item====
; SubItems use the MenuItem structure.
Pen1SubItem dc.l Pen2SubItem
            dc.w 65,0,100,10
            dc.w $157          ;Flags
            dc.l $FFFFFFFE     ;Exclude all other items linked to this one
            dc.l Pen1Text,0
            dc.b '1'           ;Command
            dc.b 0
            dc.l 0
            dc.w 0
Pen2SubItem dc.l Pen3SubItem
            dc.w 65,10,100,10,$57
            dc.l $FFFFFFFD     ;exclude other subitems
            dc.l Pen2Text,0
            dc.b '2',0
            dc.l 0
            dc.w 0
Pen3SubItem dc.l 0                ;no more SubItems under Color
            dc.w 65,20,100,10,$57
            dc.l $FFFFFFFB
            dc.l Pen3Text,0
            dc.b '3',0
            dc.l 0
            dc.w 0
;==========IntuiText Structures for Menu0========
InfoText dc.b  0            ;FrontPen
         dc.b  1            ;BackPen
         dc.b  1            ;Draw Mode = JAM2
         dc.b  0            ;Pad byte
         dc.w  0            ;LeftEdge
         dc.w  0            ;TopEdge
         dc.l  TextAttr     ;TextFont
         dc.l  InfoString   ;ptr to String
         dc.l  0            ;NextText, if any
QuitText  dc.b 0,1,1,0
          dc.w 0,0
          dc.l TextAttr,QuitString,0
ColorText dc.b 0,1,1,0
          dc.w 0,0
          dc.l TextAttr,ColorString,0
Pen1Text  dc.b 0,1,1,0
          dc.w 19          ;allow room for the checkmark
          dc.w 0
          dc.l TextAttr,Pen1String,0
Pen2Text  dc.b 0,1,1,0
          dc.w 19,0
          dc.l TextAttr,Pen2String,0
Pen3Text  dc.b 0,1,1,0
          dc.w 19,0
          dc.l TextAttr,Pen3String,0
;========The PRINT MENU==========
PrintMenu dc.l EraseMenu
          dc.w 100,0,90,0,1
          dc.l PrintTitle
          dc.l AmazeItem
          dc.w 0,0,0,0
AmazeItem dc.l AmigaItem
          dc.w 0,0,100,10,$52
          dc.l 0,AmazeText,0
          dc.b 0,0
          dc.l 0
          dc.w 0
AmigaItem dc.l CItem
          dc.w 0,10,100,10,$52
          dc.l 0,AmigaText,0
          dc.b 0,0
          dc.l 0
          dc.w 0
CItem     dc.l 0
          dc.w 0,20,100,10,$52
          dc.l 0,CText,0
          dc.b 0,0
          dc.l 0
          dc.w 0
AmazeText dc.b 0,1,1,0
          dc.w 0,0
          dc.l TextAttr,AmazeString,0
AmigaText dc.b 0,1,1,0
          dc.w 0,0
          dc.l TextAttr,AmigaString,0
CText     dc.b 0,1,1,0
          dc.w 0,0
          dc.l TextAttr,CString,0
;========THE ERASE MENU=========
; Since the erase menu's items are exactly the same as the print menu
; except for the initial menu position, use the same Item structures.
EraseMenu dc.l 0
          dc.w 200,0    ;Note the different X position than Print menu.
          dc.w 90,0,1
          dc.l EraseTitle
          dc.l AmazeItem
          dc.w 0,0,0,0
MyWindow       dc.l 0
RastPort       dc.l 0
_IntuitionBase dc.l 0
_GfxBase       dc.l 0
```



```
MainTaskAddr  dc.l 0
FontPtr    dc.l 0          ;where to store the address of the Font
;=====Text Attribute Structure======
TextAttr   dc.l FontName   ;address of 'topaz.font',0
FontHeight dc.w 8          ;desired height of Font (ySize)
           dc.b 0          ;FC_STYLE = NORMAL
           dc.b 0          ;FC_FLAGS = FPB_ROMFONT
;======The Message Structures for 5 Strings=======
; These structures are my own creation, not Amiga system structs.
MsgStruct  dc.l AmazingMess ;the address of String #1
           dc.w 150,50        ;the X, Y co-ordinates of String #1
           dc.l AmigaMess    ;the address of String #2
           dc.w 150,100       ;the X, Y co-ordinates of String #2
           dc.l CMess
           dc.w 150,150
           dc.l Explain1Mess
           dc.w 95,30
           dc.l Explain2Mess
           dc.w 95,40
Quit          dc.b  1     ;When this is set to 0, we exit the program.
PrintMode     dc.b  1     ;1 for Print, 0 for Erase.
Title:
  dc.b '   AMIGA ASSEMBLY EXAMPLE (1.0) by Jeff Glatt JULY 88',0
ProjectTitle  dc.b 'Project',0
PrintTitle    dc.b 'Print',0
EraseTitle    dc.b 'Erase',0
QuitString    dc.b 'Quit',0
CString       dc.b 'Subvert',0
AmazeString   dc.b 'Amazing',0
AmigaString   dc.b 'Amiga',0
InfoString    dc.b 'Info',0
ColorString   dc.b 'Color',0
FontName      dc.b 'topaz.font',0
IntuitionName dc.b 'intuition.library',0
GfxName       dc.b 'graphics.library',0
AmazingMess   dc.b 'Buy Amazing Computing',0
AmigaMess     dc.b 'Deputy Dan has no friends.',0
CMess         dc.b 'Throw away your C compilers today!',0
Pen1String    dc.b 'Pen 1',0
Pen2String    dc.b 'Pen 2',0
Pen3String    dc.b 'Pen 3',0
  ;The following string of Spaces is for erasing purposes.
Spaces        dc.b '                                                ',0
Explain1Mess  dc.b 'This is an assembly language example of how to',0
Explain2Mess  dc.b 'use Amiga window, menu, and intuition functions.',0
  END
```

•AC•

AmigaDOS, Assembly Language, and FileNotes

Programming Your Amiga

by Dan Huth

Accurate, descriptive file naming is one of our weapons in the war against information overload. Unfortunately, disk filenames have often been limited to ridiculously short lengths: computer users constantly try to cram accurate file descriptions into as few as eight characters. But systems programmers are lately becoming more concerned about such things, and have been designing disk operating systems that allow lengthy, even verbose, filenames. Witness my trusty old Commodore-64 (vintage 1983) and its generous 16-character filenames. However, the doors to file naming verbosity didn't open for me until March 1986, when I became the proud owner of a sparkling new Amiga 1000 that boasted (Amigas are very proud machines) huge thirty-character filenames.

Thirty characters are an awful lot. For example, Assembler_Macros_&_Constants.i is a legal filename under AmigaDOS. Pretty descriptive, eh? But there's more. For those situations in which even a thirty-character filename isn't long enough, AmigaDOS provides its users with the ability to append an eighty-character comment to any disk file via the FileNote command. To add a filenote to a disk file named WarmFuzzyData in the UFO directory on the disk in my external drive, I'd use this command:

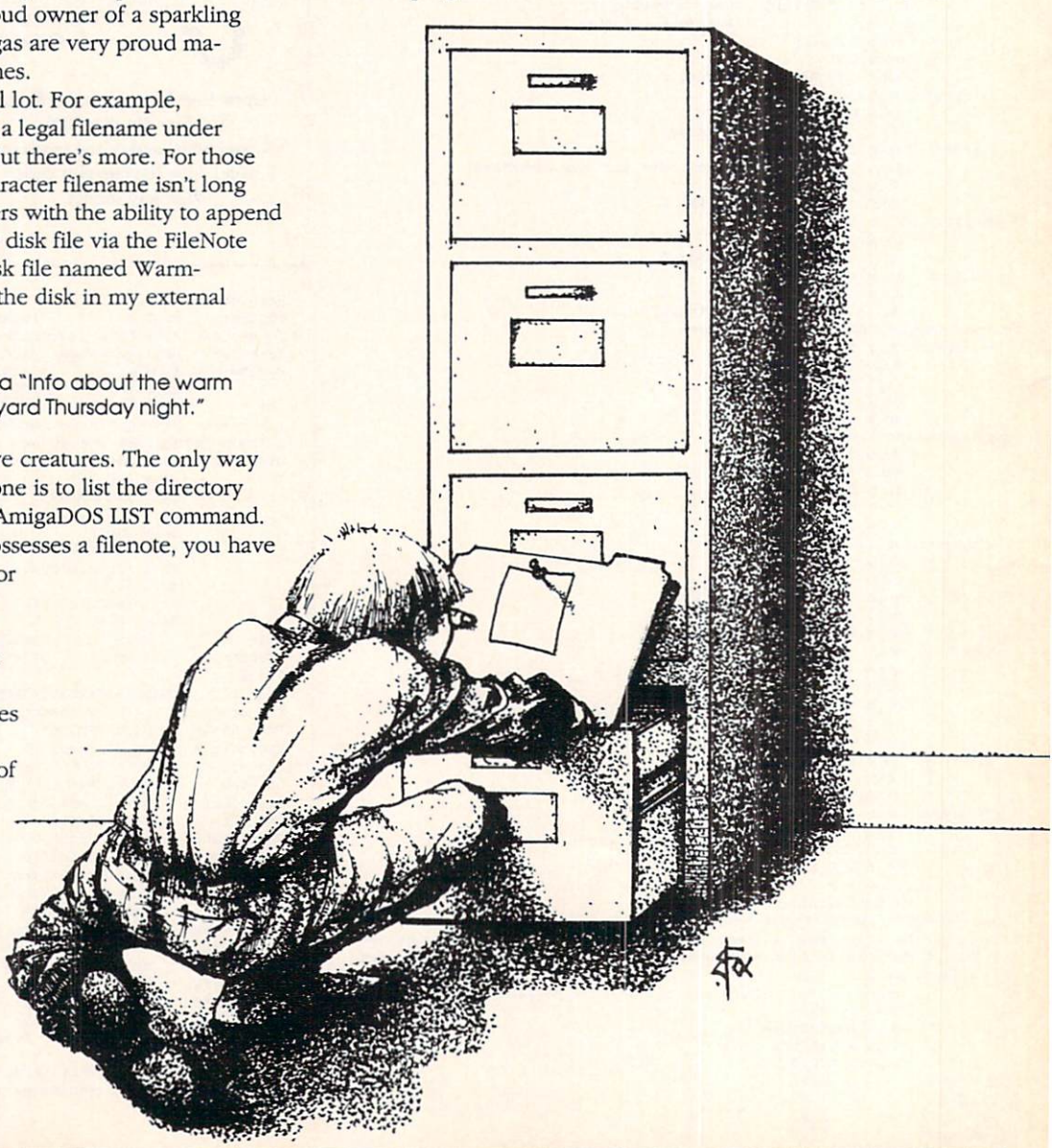
```
1> FileNote DF1:UFO/WarmFuzzyData "Info about the warm fuzzy things that landed in the backyard Thursday night."
```

But filenotes are rather elusive creatures. The only way to find out if a directory possesses one is to list the directory containing that directory, using the AmigaDOS LIST command. To determine if an individual file possesses a filenote, you have the option of listing either that file, or the directory containing it. Thus, to read all the filenotes on a particular disk, the poor AmigaDOS user must list each directory on that disk. For example, to find out if any of the files on a standard Workbench1.2 disk contain filenotes, I'd use this series of commands:

```
list Workbench1.2:
list Workbench1.2:Trashcan
list Workbench1.2:c
list Workbench1.2:Demos
list Workbench1.2:System
list Workbench1.2:l
list Workbench1.2:devs
list Workbench1.2:devs/keymaps
list Workbench1.2:devs/printers
list Workbench1.2:devs/clipboards
```

and so on. (Fourteen more list commands would be required to complete my search.) As you can see, accessing a filenote isn't as easy as accessing a filename. (The names of all the files on the disk would be displayed if I used the command DIR Workbench 1.2: opt a.) I suspect that, because of their relative inaccessibility, filenotes are used less often than they should be.

If filenotes were more accessible, people would be more apt to use them. Very late one night I started feeling sorry for the poor things. I decided to become a filenote partisan, and wrote a program that would free all those forlorn filenotes out



there from their dark jail cells of obscurity and let them in the bright sunshine of prominence. In other words, I wrote a program that can search an entire disk, and display any filenotes it finds.

How to use ShowFileNotes

The program is ShowFileNotes. Its I/O is like this:

```
1> ShowFileNotes Work:
Work:Test/Test2 (DIR)
 This is a test subdirectory.
Work:Test/Test2/Test3/Datefile (FILE)
 This file has absolutely no significance. It is a test file only.
Work:Source/PopToFront.a (FILE)
 From Transactor
Work:Source/Header.a (FILE)
 Skeleton for assembly language source files
Work:Source/TestForm (FILE)
 Test form for EmergencyDB
1>
```

Note that ShowFileNotes accepts a command line argument. Only the first argument is useful, since ShowFileNotes ignores anything on the command line after the first argument. A quoted argument is allowed, and must be used if the file or directory name specified contains spaces. If no argument is specified, then ShowFileNotes uses the current directory, so that the command ShowFileNotes Df1: is equivalent to the command CD Df1: followed by ShowFileNotes. If the directory or file given as an argument can't be found, ShowFileNotes displays an error message, then exits.

How ShowFileNotes Works

(Note: since I need to use the phrase "file or directory" so often, I'm going to substitute the term "object" for that phrase in the remainder of this article.)

ShowFileNotes's job is to check each object in a given directory and all its subdirectories, searching for any objects that have filenotes specified for them. If any such objects are found, both the object's name and its associated filenote are displayed in the AmigaDOS window.

Filenotes are stored on the disk in the object's file header block, but fortunately there's no need to actually read the file headers from the disk to spot objects that possess filenotes. Instead, we can use the DOS library functions Examine() and ExNext(), which take care of the low-level disk accessing for us.

Examine() is a function that takes a lock and a pointer to a block of memory called a FileInfoBlock (FIB) as its arguments, and copies information from the disk into the FIB concerning the object associated with the lock. We obtain a lock by using the DOS library function Lock(), and we receive a pointer to a FIB when we call the Exec library function AllocMem(). The assembly language code fragment below illustrates the process.

```
;— Obtain a Lock
move.l  a3,d1                   ;assuming directory name is at
(a3)
moveq   #ACCESS_READ,d2         ;we only want to read the object
jsr     _LVOLock(a4)            ;assuming DOSBase is in a4
move.l  d0,d7                   ;the Lock is now in d7
beq     Finished                ;Lock() returns 0 if it fails

;— Obtain a FileInfoBlock
move.l  #fib_SIZEOF,d0          ;size of FIB in bytes
move.l  #MEMF_CLEAR,d1          ;zero the memory region
jsr     _LVOAllocMem(a6)        ;assuming ExecBase is in a6
```



```
tst.l   d0                      ;AllocMem() returns 0 if it fails
beq     Finished
movea.l d0,a2                   ;the FIB pointer is now in a2

;— Finally call Examine
move.l  d7,d1                   ;directory lock
move.l  a2,d2                   ;FIB pointer
jsr     _LVOExamine(a4)         ;DOSBase is still in a4
```

Now that we have all the relevant information about the object in our FIB, we can access it in this manner:

```
;— Determine whether the object is a file or a directory
tst.l   fib_DirEntryType(a2) ;the FIB pointer is still in a2
bgt     Directory            ;if it's positive, it's a directory
blt     File                 ;if it's negative, it's a file

;— Determine if the object possesses a filenote
tst.b   fib_Comment(a2)      ;the FIB pointer is still in a2
beq     NoComment            ;if it's zero, there's no filenote
bne     Comment              ;if it's nonzero, there's a
filenote

;— Get a pointer to the object's name
lea     fib_FileName(a2),a0  ;pointer to filename is now in a0
move.l  a0,d1                ;now the pointer is also in d1
```

The first thing we should check is fib_DirEntryType, to find out whether the object we've obtained a lock on is a file or a directory. If it's a file, ShowFileNotes will look only at this file, and nothing else on the disk. If, however, it's a directory, we'll need to look at each object in it.

(continued)

We can look at each object in a directory by calling ExNext(). ExNext() is designed to be called after Examine(), but only if the object examined turned out to be a directory. ExNext() returns information in the FIB about the next entry in that directory. And, succeeding calls to ExNext() return information about succeeding entries in the directory, until the last entry in the directory is encountered. At that time ExNext() returns a value of 0, telling us there are no more entries in this directory. (Actually, we can't be sure that the error resulted from reaching the end of the directory, unless we call the DOS library function IoErr() immediately after receiving the error return value from ExNext(). In such a case, if IoErr() returns the value ERROR_NO_MORE_ENTRIES we can be sure. ShowFileNotes isn't this fussy; it assumes that an error return from ExNext() is due to a "no more entries" condition.)

In pseudocode, the process of examining the contents of a directory is as follows:

```
Label1:
Obtain Lock and FIB for the filing system object in question
Call Examine()
   ( — Process data returned in FIB — )
If (the object is not a directory) then branch to Label2
Call ExNext()
While (ExNext() didn't return zero)
   ( — Process data returned in FIB — )
   If (this directory entry is a subdirectory) then
      Save necessary register values
      Perform recursive subroutine branch to Label 1
       Restore register values
   End If
   Call ExNext()
End While
Label2:
Release Lock and deallocate FIB
Return from subroutine
```

The careful reader may have noticed that the above pseudo-code is general in nature; the two lines of (— Process data returned in FIB —) are where all of ShowFileNotes's specific code goes. In fact, just about any program that needs to examine the entries of disk directories, including nested subdirectories, can use the above code skeleton. I imagine about an hour's work would suffice to transform ShowFileNotes into a program displaying a disk directory a la either of the AmigaDOS commands DIR or LIST.

Since the source code for ShowFileNotes is included with this article, you can examine it to gain further insights concerning the interaction between AmigaDOS and the assembly language programmer. In particular, the sections of code that deal with the manipulation of file and directory names may be of interest to you, as they illuminate the handling of deeply-nested directory trees: ShowFileNotes will without a gulp tell you that a certain file named

Work:test/test2/test3/test4/test5/test6/test7/test8/datefile

has as a filenote

Hi. Didn't think you'd find me.

One last note: There's no error message generation in ShowFileNotes. This would be a worthy programming exercise for anyone who wants to sharpen his or her Amiga assembly language programming skills; it won't be difficult, and it will be instructive.

Listing One:

```
**********************************************
* ShowFileNotes.a  —  displays filenote &
* filename for all files in a given directory
* (and all subdirectories thereof) that have
* filenotes attached to them.
*
* Version 0.08    10 May 88
* Copyright (C)   Dan Huth  P.O. Box 402
*                 Bolivar, OH 44612
*
* Permission is hereby granted for unrestricted
* use of the accompanying source code, and
* executable code resulting therefrom, with the
* provision that this header must remain intact
* in any distribution of the source code, and
* must accompany (in human-readable form) any
* distribution of executable code resulting
* therefrom.
**********************************************

        nolist
        include "exec/types.i"
        include "exec/memory.i"
        include "libraries/dosextens.i"
        list

CALL    macro
        jsr     _LVO\1(a6)
        endm
DOSCALL macro
        jsr     _LVO\1(a4)
        endm
XLIB    macro
        xref    _LVO\1
        endm
RETURN  macro
        rts
        endm

        xref    _AbsExecBase
        xref    _printf
        XLIB    OpenLibrary
        XLIB    FindTask
        XLIB    AllocMem
        XLIB    FreeMem
        XLIB    Lock
        XLIB    DupLock
        XLIB    CurrentDir
        XLIB    UnLock
        XLIB    Examine
        XLIB    ExNext
        XLIB    Output

        xdef    _DOSBase     ;exports for _printf
        xdef    _stdout

MAXLINELEN equ  64           ;LONGs
ERROR   equ     1
```

```
*************************************************
* Main program section
*
* Register use: a0-a1/d0-d2 are scratch
* a2 FIB pointer       d3 #fib_SIZEOF
* a3 ArgBuffer pointer d4 Old curreny dir'y lock
* a4 _DOSBase          d5 error flag
* a5 frame pointer     d6 NULL byte in ArgBuffer
* a6 ExecBase          d7 argc / directory lock
*
* _DOSBase must be in a4 for DOSCALL to work.
*************************************************

Main    link    a5,#-(MAXLINELEN*4)
        movea.l SP,a3         ;a3 == &ArgBuffer
        bsr     ProcCmdLine   ;returns 0 if
        tst.l   d0            ;  successful
        bne.s   Finished      ;exit-cmd line err
        move.l  #fib_SIZEOF,d3  ;for efficiency

 ;-Open DOS library (d0 must be zero)
        movea.l _AbsExecBase,a6  ;using Exec lib
        lea     DosName(PC),a1
           ;any version (d0 already holds zero)
        CALL    OpenLibrary
        movea.l d0,a4         ;a4 == _DOSBase
        move.l  d0,_DOSBase   ;for _printf

 ;-Obtain standard output file handle
        DOSCALL Output
        move.l  d0,_stdout    ;for _printf

****************
* Obtain lock on directory. If a dir'y name was
* specified on the cmd line, try to lock that
* dir'y. Otherwise, use the current directory.
****************

        tst.l   d7            ;d7 == argc
        bne.s   TryToLock     ;argc > 0 - branch

 ;-D7 holds dir'y lock, not argc, from now on
 ;-Argc == 0; use CLI's current directory
        suba.l  a1,a1         ;find this task
        CALL    FindTask
        movea.l d0,a2         ;process struct ptr
        move.l  pr_CurrentDir(a2),d1
        DOSCALL DupLock       ;dup curr dir lock
        move.l  d0,d7         ;d7 == dir lock
        movea.l pr_CLI(a2),a1  ;CLI struct BPTR
        adda.l  a1,a1         ;BPTR conversion
        adda.l  a1,a1         ;now it's a pointer
        movea.l cli_SetName(a1),a1  ;cur dir BSTR
        adda.l  a1,a1         ;BPTR conversion
        adda.l  a1,a1         ;dir'y length/name
        move.b  (a1)+,d0      ;d0.b holds length
        ext.w   d0            ;convert to word
        movea.l a3,a0         ;a0 == &ArgBuffer

 ;-Move current dir'y name into ArgBuffer
 ;-a1 points to dir'y name, a0 to ArgBuffer,
 ;-d0 holds string length, d6 will index NULL
 ;-(d6 was zeroed by ProcCmdLine.)
        subq.w  #1,d0         ;dbra quits on -1
MoveCDName
        move.b  (a1)+,(a0)+
        addq.w  #1,d6         ;keep track of NULL
        dbra    d0,MoveCDName
        clr.b   (a0)          ;NULL termination
        bra.s   Finally       ;0(a3,d6.w) == NULL

 ;-Argc == 1; try to lock directory (or file)
 ;-specified in command line
TryToLock
        move.l  a3,d1         ;directoryy name
        moveq   #ACCESS_READ,d2
        DOSCALL Lock
        move.l  d0,d7         ;d7 == dir'y lock
        bne.s   Finally       ;returns 0 for fail
        moveq   #ERROR,d0
```

(continued)

```
        bra.s   Finished      ;exit on error

****************
* FINALLY THE MEAT AND POTATOES
****************

Finally bsr.s   MeatAndSpuds
Finished
        unlk    a5
        moveq   #0,d0
        RETURN

****************
* SUBROUTINES
****************

 ;-Allocate a FileInfoBlock (FIB)
MeatAndSpuds
        move.l  d3,d0         ;d3 == #fib_SIZEOF
        move.l  #MEMF_CLEAR,d1
         CALL    AllocMem
         tst.l   d0            ;returns 0 for fail
        beq     EndMAS2       ;exit on failure
        movea.l d0,a2         ;a2 == &FIB

 ;-Examine this thing
        move.l  d7,d1         ;directory lock
        move.l  a2,d2         ;&FIB
        DOSCALL Examine
        tst.l   d0            ;returns 0 for fail
        beq     EndMAS        ;exit on failure
        tst.l   fib_DirEntryType(a2)
        bgt.s   Directory
        bsr     File
        bra     EndMAS

 ;-This thing is a directory
 ;-Make it the current directory
Directory
        move.l  d7,d1         ;d1 = d7 = dir lock
        DOSCALL CurrentDir
        move.l  d0,d4         ;d4 == Old CD lock

 ;-If it has a filenote, display it
        tst.b   fib_Comment(a2)
        beq.s   ExamNext      ;no filenote
        pea     fib_Comment(a2)
        move.l  a3,-(SP)      ;directory name
        pea     DFormatStr(PC)
        jsr     _printf
        lea     12(SP),SP     ;pop three arguments

 ;-Examine entries in this directory
ExamNext
        move.l  d7,d1         ;directory lock
        move.l  a2,d2         ;&FIB
        DOSCALL ExNext
        tst.l   d0            ;returns 0 on fail
        beq.s   EndDirectory ;no more entries
        tst.l   fib_DirEntryType(a2)
        bgt.s    SubDirectory
        bsr.s    File
        bra.s   ExamNext

 ;-Handle subdirectory
SubDirectory
        movem.l a2/d4/d6-d7,-(SP)
        lea     fib_FileName(a2),a0
        move.l  a0,d1
        moveq   #ACCESS_READ,d2
        DOSCALL Lock          ;Lock this subdir'y
        move.l  d0,d5         ;ret val from Lock()
        move.l  d0,d7         ;subdirectory lock
        beq.s   EndSubDir     ;returns 0 for fail

 ;-Build new directory path
        lea     fib_FileName(a2),a0  ;filename
        lea     0(a3,d6.w),a1  ;NULL position
        cmpi.b  #':',-1(a1)
        beq.s   2$
        move.b  #'/',(a1)+
1$      addq.w  #1,d6         ;keep track of NULL
2$      move.b  (a0)+,(a1)+   ;copy
        bne.s   1$            ;stop after NULL

 ;-Recurse
        bsr     MeatAndSpuds ;RECURSION
EndSubDir
        movem.l (SP)+,a2/d4/d6-d7
        tst.l   d5            ;Lock() error?
        beq.s   EndDirectory ;yes - exit
        clr.b   0(a3,d6.w)    ;restore NULL byte
        bra.s   ExamNext

 ;-Housekeeping
EndDirectory
        move.l  d4,d1         ;restore orig'l cd
        DOSCALL CurrentDir
EndMAS  movea.l a2,a1         ;&FIB
        move.l  d3,d0         ;#fib_SIZEOF
         CALL    FreeMem       ;deallocate FIB
EndMAS2                       ;Fall through

UnLockSbr
        move.l  d7,d1         ;directory lock
        DOSCALL UnLock
        RETURN

 ;-If this file has a filenote, display it
File    tst.b   fib_Comment(a2)
        beq.s   EndFile       ;no filenote
        pea     fib_Comment(a2)
        pea     fib_FileName(a2)  ;file name
        cmpi.b  #':',-1(a3,d6.w)  ;in root dir
        bne.s   1$            ;no - branch
        pea     ColonStr(PC) ;yes
        bra.s   2$
1$      pea     SlashStr(PC) ;no
2$      move.l  a3,-(SP)      ;directory name
        pea     FFormatStr(PC)
        jsr     _printf
        lea     20(SP),SP     ;pop five arguments
EndFile RETURN

*************************************************
* ProcCmdLine subroutine
*
* Enter with cmd line address in a0, cmd line
* length in d0; exit with error code (0 for no
* error) in d0, argc in d7, position of
* ArgBuffer NULL byte in d6, and (one only) arg
* copied into ArgBuffer if any cmd line args
* were found.
*
* Register use:
```

```
*  a0 pointer to next byte in command line
*  a1 pointer to next byte in ArgBuffer
*  d0 bytes remaining in command line
*  d1 current char from command line
*  d6 position of terminating NULL in ArgBuffer
*  d7 argc
*********************************************

ProcCmdLine
        moveq   #0,d7           ;d7 = argc
        moveq   #0,d6           ;d6 = NULL position
        movea.l a3,a1           ;&ArgBuffer

 ;-Strip off ASCII chars < 33 or > 127
GetArg  move.b  (a0)+,d1
        subq.w  #1,d0
        ble.s   ExitPCL         ;exit - no argument
        cmpi.b  #' ',d1
        ble.s   GetArg

 ;-Check for quoted argument
        cmpi.b  #'"',d1
        bne.s   3$
        bra.s   2$

 ;-Process quoted argument
1$      cmpi.b  #'"',d1
        beq.s   EndGetArg       ;exit (discard " )
        move.b  d1,(a1)+
        addq.w  #1,d6           ;track NULL
2$      move.b  (a0)+,d1        ;enter (discard " )
        subq.w  #1,d0
        bgt.s   1$
        moveq   #ERROR,d0       ;no trailing quote
        bra.s   ExitPCL         ;exit on error

 ;-Process unquoted argument
3$      move.b  d1,(a1)+
        addq.w  #1,d6           ;track NULL
        move.b  (a0)+,d1
        subq.w  #1,d0
        cmpi.b  #' ',d1
        bgt.s   3$

 ;-Handle end of argument
EndGetArg
        addq.w  #1,d7           ;return argc in d7,
                                ;NULL position in d6,
        moveq   #0,d0           ;success code in d0
        clr.b   (a1)            ;null-termination
ExitPCL RETURN

*********************************************
* data storage declarations
*********************************************

        ;NOTE: after DosName is used to open the
        ;DOS library, the 12 bytes occupied by
        ;that string aren't accessed at all; I
        ;use them to store two LONG variables.

_DOSBase
DosName dc.b    'dos.'
_stdout dc.b    'libr'
        dc.b    'ary',0
DFormatStr
        dc.b    '%ls (DIR)',13,10
        dc.b    '  %ls',13,10,0
        cnop    0,2
FFormatStr
        dc.b    '%ls%ls%ls (FILE)',13,10
        dc.b    '  %ls',13,10,0
        cnop    0,2
SlashStr
        dc.b    '/',0
ColonStr
        dc.b    0

        end
```

•AC•

The AMICUS & Fred Fish Public Domain Software Library

This software is collected from user groups and electronic bulletin boards around the nation. Each Amicus disk is nearly full, and is fully accessible from the Workbench. If source code is provided for any program, then the executable version is also present. This means that you don't need the C compiler to run these programs. An exception is granted for those programs only of use to people who own a C compiler.

The Fred Fish disk are collected by Mr. Fred Fish, a good and active friend of the Amiga.

Note: Each description line below may include something like 'S-O-E-D', which stands for 'source, object file, executable and documentation'. Any combination of these letters indicates what forms of the program are present. Basic programs are presented entirely in source code format.

AMICUS Disk 1

ABasic programs: Graphics

3DSolids	3d solids modeling prog. w/sample data files
Blocks	draws blocks
Cubes	draws cubes
Durer	draws pictures in the style of Durer
FScape	draws tractal landscapes
Hidden	3D drawing program, w/ hidden line removal
JPad	simple paint program
Optical	draw several optical illusions
PaintBox	simple paint program
Shuttle	draws the Shuttle in 3d wireframe
SpaceArt	graphics demo
Speaker	speech utility
Sphere	draws spheres
Spiral	draws color spirals
ThreeDee	3d function plots
Topography	artificial topography
Wheels	draws circle graphics
Xenos	draws fractal planet landscapes

ABasic programs: Tools

AddressBook	simple database program for addresses
CardFile	simple card file database program
Demo	multiwindow demo
KeyCodes	shows keycodes for a key you press
Menu	run many ABasic programs from a menu
MoreColors	way to get more colors on the screen at once, using aliasing
shapes	simple color shape designer Speakit speech and narrator demo

ABasic programs: Games

BrickOut	classic computer brick wall game
Othello	also known as 'go'
Saucer	simple shoot-em-up game
Spelling	simple talking spelling game
ToyBox	selectable graphics demo

ABasic programs: Sounds

Entertainer	plays that tune
HAL9000	pretends it's a real computer
Police	simple police siren sound
SugarPlum	plays "The Dance of the Sugarplum Fairies"

C programs:

ATerm	simple terminal program, S-E
cc	aid to compiling with Lattice C
decvnt	opposite of CONVERT for cross developers
Dotty	source code to the 'dotty window demo
echox	unix-style filename expansion, partial S,O-D
fasterfp	explains use of fast-floating point math
FixDate	fixes future dates on all files on a disk, S-E
freedraw	simple Workbench drawing prog.,S-E
GfxMem	graphic memory usage indicator, S-E
Grep	searches for a given string in a file with
docs.	ham shows off the hold-and-modify method of color generation
IBM2Amiga	fast parallel cable transfers between an IBM and an Amiga
Mandel	Mandelbrot set program, S-E
moire	patterned graphic demo, S-E
objfix	makes Lattice C object file symbols visible to Wack, S-E
quick	quick sort strings routine
raw	example sample window I/O
setlace	turns on interlace mode, S-E
sparks	qix-type graphic demo, S-E

Other executable programs:

SpeechToy	speech demonstration
WhichFont	displays all available fonts

Texts:

68020	describes 68020 speedup board from CSA
Aliases	explains uses of the ASSIGN command
Bugs	known bug list in Lattice C 3.02
CLICard	reference card for AmigaDOS CLI
CLICommands	guide to using the CLI
Commands	shorter guide to AmigaDOS CLI commands
EdCommands	guide to the ED editor
Filenames	AmigaDOS filename wildcard conventions
HalfBright	explains rare graphics chips that can do more colors
ModemPins	description of the serial port pinout
RAMdisks	tips on setting up your RAM: disk
ROMWack	tips on using ROMWack
Sounds	explanation of Instrument demo sound file format
Speed	refutation of Amiga's CPU and custom chip speed
WackCmds	tips on using Wack

AMICUS Disk 2

C programs:

alib	AmigaDOS object library manager , S-E
ar	text file archive program, S-E
fixobj	auto-chops executable files
shell	simple CLI shell, S-E
sq, usq	file compression programs, S-E
YachtC	a familiar game, S-E
Make	a simple 'make' programming utility, S-E
Emacs	an early version of the Amiga text editor, S-E-D

Assembler programs:

bsearch.asm	binary search code
qsort.asm	Unix compatible qsort() function, source and C test program
setjmp.asm	setjmp() code for Lattice 3.02
SVprintf	Unix system V compatible printf()
trees.o	Unix compatible tree() function, O-D

(This disk formerly had IFF specification files and examples. Since this spec is constantly updated, the IFF spec files have been moved to their own disk in the AMICUS collection.)

John Draper Amiga Tutorials:

Animate	describes animation algorithms
Gadgets	tutorial on gadgets
Menus	learn about Intuition menus

AMICUS Disk 3

C programs:

Xref	a C cross-reference gen., S-E
6bitcolor	extra-half-bright chip gfx demo, S-E
Chop	truncate (chop) files down to size, S-E
Cleanup	removes strange characters from text files
CR2LF	converts carriage returns to line feeds in Amiga files, S-E
Error	adds compile errors to a C file, S
Hello	window ex. from the RKM, S
Kermit	generic Kermit implementation, flakey, no terminal mode, S-E
Scales	sound demo plays scales, S-E
SkewB	Rubik cube demo in hi-res colors, S-E

AmigaBasicProgs(dir)

Automata	cellular automata simulation
CrazyEights	card game
Graph	function graphing programs
WitchingHour	a game

ABasiC programs:

Casino	games of poker, blackjack, dice, and craps
Gomoku	also known as 'othello'
Sabotage	sort of an adventure game

Executable programs:

Disassem	a 68000 disassembler, E-D
DpSlide	shows a given set of IFF pictures, E-D
Arrange	a text formatting program, E-D

Assembler programs:

Argoterm	terminal program with speech and Xmodem, S-E

AMICUS Disk 4 Files from the original Amiga Technical BBS

Note that some of these files are old, and refer to older versions of the operating system. These files came from the Sun system that served as Amiga technical support HQ for most of 1985. These files do not carry a warranty, and are for educational purposes only. Of course, that's not to say they don't work.

Complete and nearly up-to-date C source to 'image.ed', an early version of the Icon Editor. This is a little flaky, but compiles and runs.

An Intuition demo, in full C source, including files: demomenu.c, demomenu2.c, demoreq.c, getascii.c, idemo.c, idemo.guide, idemo.make, idemoall.h, nodos.c, and txwrite.c

addmem.c	add external memory to the system
bobtest.c	example of BOB use
consoleIO.c	console IO example
creaport.c	create and delete ports
creastdi.c	create standard I/O requests
creatask.c	creating task examples
diskio.c	example of track read and write
dotty.c	source to the 'dotty window' demo
dualplay.c	dual playfield example
flood.c	flood fill example
freemap.c	old version of 'freemap'
gelltools.c	tools for VSprites and BOBs
gfxmem.c	graphic memory usage indicator
hello.c	window example from RKM
inputdev.c	adding an input handler to the input stream
joystik.c	reading the joystick
keybd.c	direct keyboard reading
layertes.c	layers examples
mousport.c	test mouse port
ownlib.c, ownlib.asm	example of making your own library with Lattice
paratest.c	tests parallel port commands
seritest.c	tests serial port commands
serisamp.c	example of serial port use
prinintr.c	sample printer interface code
prtbase.h	printer device definitions
regintes.c	region test program
setlace.c	source to interlace on/off program
setparallel.c	set the attributes of the parallel port
SetSerial.c	set the attributes (parity, data bits, etc.) of the serial port
singplay.c	single playfield example
speechtoy.c	source to narrator and phonetics demo
timedely.c	simple timer demo
timer.c	exec support timer functions
timrstuf.c	more exec support timer functions
WhichFont.c	loads and displays all available system fonts

process.i and prtbase.i assmebler include files:

autorqstr.txt	warnings of deadlocks with autorequesters
consoleIO.txt	copy of the RKM console I/O chapter
diskfont.txt	warning of disk font loading bug
fullfunc.txt	list of #defines, macros, functions
inputdev.txt	preliminary copy of the input device chapter

License information on Workbench distribution license

printer pre-release copy of the chapter on printer drivers, from RKM 1.1 v11fd.txt 'diff' of .fd file changes from version 1.0 to 1.1 v28v1.diff 'diff' of include file changes from version 28 to 1.0

AMICUS Disk 5 Files from the Amiga Link / Amiga Information Network

Note that some of these files are old, and refer to older versions of the operating system. These files are from Amiga Link. For a time, Commodore supported Amiga Link, aka AIN, for online developer technical support. It was only up and running for several weeks. These files do not carry a warranty, and are for educational purposes only. Of course, that's not to say they don't work.

A demo of Intuition menus called 'menudemo', in C source

whereis.c	find a file searching all subdirectories
bobtest.c	BOB programming example
sweep.c	sound synthesis example

Assembler files:

mydev.asm	sample device driver
mylib.asm	sample library example
mylib.i	
mydev.i	
asmsupp.i	
macros.i	assembler include files

Texts:

amigatricks	tips on CLI commands
extdisk	external disk specification
gameport	game port spec
parallel	parallel port spec
serial	serial port spec
v1.1update	list of new features in version 1.1
v1.1h.txt	'diff' of include file changes from version

Files for building your own printer drivers, including dospecial.c, epsondata.c, init.asm, printer.c, printer.link, printertag.asm, render.c, and wait.asm. This disk does contain a number of files describing the IFF specification. These are not the latest and greatest files, but remain here for historical purposes. They include text files and C source examples. The latest IFF spec is elsewhere in this library.

AMICUS Disk 6 IFF Pictures

This disk includes the DPSlide program, which can view a given series of IFF pictures, and the 'showpic' program, which can view each file at the click of an icon. The pictures include a screen from ArticFox, a Degas dancer, the guys at Electronic Arts, a gorilla, horses, King Tut, a lighthouse, a screen from Marble Madness, the Bugs Bunny Martian, a still from an old movie, the Dire Straits moving company, a screen from Pinball Contruction Set, a TV newcaster, the PaintCan, a world map, a Porsche, a shuttle mission patch, a tyrannosaurus rex, a planet view, a VISA card, and a ten-speed.

AMICUS Disk 7 DigiView HAM demo picture disk

This disk has pictures from the DigiView hold-and-modify video digitizer. It includes the ladies with pencils and lollypops, the young girl, the bulldozer, the horse and buggy, the Byte cover, the dictionary page, the robot and Robert. This includes a program to view each picture separately, and all together as separate, slidable screens. The 'seeilbm' program, to turn any screen into an IFF picture.

AMICUS Disk 8

C programs:

Browse	view text files on a disk, using menus S-E-D
Crunch	removes comments and white space from C files, S-E
IconExec	EXECUTE a series of commands from Workbench S-E
PDScreen Dump	dumps Rastport of highest screen to printer
SetAlternate	sets a second image for an icon, when clicked once S-E
SetWindow	makes windows for a CLI program to run under Workbench S-E
SmallClock	a small digital clockin a window menu bar
Scrimper	the screen printer in the fourth AC S-E

Amiga Basic Programs:

(Note: Many of these programs are present on AMICUS Disk 1. Several of these were converted to Amiga Basic, and are included here.)

AddressBook	a simple address book database
Ball	draws a ball
Cload	program to convert Compuserve hex files to binary, S-D
Clue	the game, Intuition driven
ColorArt	art drawing program
DeluxeDraw	the drawing program in the 3rd AC, S-D
Eliza	conversational computer psychologist
Othello	the game, as known as 'go'
RatMaze	3D ratmaze game
ROR	boggling graphics demo
Shuttle	draws 3D pictures of the space shuttle
Spelling	simple spelling program
YoYo	wierd zero-gravity yo-yo demo, tracks yo-yo to the mouse

Executable programs:

3Dcube	Modula-2 demo of a rotating cube
Alticon	sets a second icon image, displayed when the icon is clicked
AmigaSpell	a slow but simple spell checker, E-D
arc	the ARC file compression progra must-have for telecom, E-D
Bertrand	graphics demo
disksalvage	prog. to rescue trashed disks, E-D
KwikCopy	a quick but nasty disk copy program: ignores errors, E-D
LibDir	lists hunks in an object file E-D
SaveILBM	saves any screen as IFF pic.E-D ??
ScreenDump	shareware screen dump prog, E only
StarTerm	version 2.0, term program, XmodemE-D

Texts:

LatticeMain	tips on fixing _main.c in Lattice
GDiskDrive	make your own 5 1/4 drive
GuruMed	explains the Guru numbers
Lat3.03bugs	bug list of Lattice C version 3.03
MForgeRev	user's view of the MicroForge HD
PrintSpooler	EXECUTE-based print spool prog.

.BMAP files:

These are the necessary links between Amiga Basic and the system libraries. To take advantage of the Amiga's capabilities in Basic, you need these files. BMAPs are included for 'clist', 'console', 'diskfont', 'exec', 'icon', 'intuition', 'layers', 'mathffp', mathieeedoubas', 'mathieeesingbas', 'mathtrans', 'potgo', 'timer' and 'translator'.

AMICUS Disk 9

Amiga Basic Programs:

FlightSim	simple flight simulator program
HuePalette	explains Hue, Saturation, & Intensity
Requester	ex. of requesters from Amiga Basic
ScrollDemo	demonstrates scrolling capabilities
Synthesizer	sound program
WorldMap	draws a map of the world

Executable programs:

Boing!	latest Boing! demo,with selectable speed.E
Brush2C	converts an IFF brush to C data instructions, initialization code, E
Brush2Icon	converts IFF brush to an icon, E
Dazzle	graphics demo, tracks to mouse, E
DeciGEL	assembler program for stopping 68010 errors, S-E-D
Klock	menu-bar clock and date display, E
life	the game of life, E
TimeSet	Intuition-based way to set the time & date
EMEmacs	another Emacs, more oriented to word processing, S-E-D
MyCLI	a CLI shell, works without the Workbench, S-E-D

Texts:

FnctnKeys	read function keys from Amiga Basic
HackerSin	explains how to win the game 'hacker'
Ist68010	guide to installing a 68010 in your Amiga
Boing!	latest Boing! demo,with selectablespeed, E
Brush2C	converts an IFF brush to C data instructions, initialization code, E
Brush2Icon	converts IFF brush to an icon, E
Dazzle	graphics demo, tracks to mouse, E
DeciGEL	assembler program for stopping 68010 errors, S-E-D
Klock	menu-bar clock and date display, E
life	the game of life, E
TimeSet	Intuition-based way to set the time date,
EMEmacs	another Emacs, more oriented to word processing, S-E-D
MyCLI	a CLI shell, works without the Workbench, S-E-D

For PDS orders, please use form on page 112

Texts:

FnctnKeys	explains how to read function keys from Amiga Basic
HackerSin	explains how to win the game 'hacker'
Ist68010	guide to installing a 68010 in your Amiga
PrinterTip	sending escape sequences to your printer
StartupTip	tips on setting up your startup-sequence file
XfrmrReview	list of Transformer programs that work

Printer Drivers:

Printer drivers for the Canon PJ-1080A, the C Itoh Prowriter, an improved Epson driver that eliminates streaking, the Epson LQ-800, the Gemini Star-10, the NEC 8025A, the Okidata ML-92, the Panasonic KX-P10xx family, and the Smith-Corona D300, with a document describing the installation process.

AMICUS Disk 10 Instrument sound demos

This is an icon-driven demo, circulated to many dealers. It includes the sounds of an acoustic guitar, an alarm, a banjo, a bass guitar, a boink, a calliope, a car horn, claves, water drip, electric guitar, a flute, a harp arpegio, a kickdrum, a marimba, a organ minor chord, people talking, pigs, a pipe organ, a Rhodes piano, a saxophone, a sitar, a snare drum, a steel drum, bells, a vibrophone, a violin, a wailing guitar, a horse whinny, and a whistle.

AMICUS Disk 11

C programs

dirutil	Intuition-based, CLI replacement manager S-E
cpri	shows and adjusts priority of CLI processes, S-E
ps	shows info on CLI processes, S-E
vidtex	displays Compuserve RLE pics, S-E

AmigaBasic programs

pointered	pointer and sprite editor program
optimize	optimization ex ample from AC article
calendar	large, animated calendar, diary and date book program
amortize	loan amortizations
brushtoBOB	converts small IFF brushes to AmigaBasic BOB OBJECTS
grids	draw and play waveforms
hilbert	draws Hilbert curves
madlib	mad lib story generator
mailtalk	talking mailing list program
meadows3D	3D graphics program, from A C™ article
mousetrack	mouse tracking example in hires mode
slot	slot machine game
tictactoe	the game
switch	pachinko-like game
weird	makes strange sounds

Executable programs

cp	unix-like copy command, E
cls	screen clear, S-E
diff	unix-like stream editor uses 'diff' output to fix files
pm	chart recorder performances indicator

Assembler programs

cls	screen clear and CLI arguments example

Modula-2

trails	moving-worm graphics demo
caseconvert	converts Modula-2 keywords to uppercase

Forth

Breshehan	circle algorithm example
Analyze	12 templates for the spreadsheet Analyze

There are four programs here that read Commodore 64 picture files. They can translate Koala Pad, Doodle, Print Shop and News Room graphics to IFF format. Getting the files from your C-64 to your Amiga is the hard part.

AMICUS Disk 12

Executable programs

blink	'alink' compatible linker, but faster, E-D
clean	spins the disk for disk cleaners, E-D
epsonset	sends Epson settings to PAR from menu E-D
showbig	view hi-res pics in low-res superbitmap, E-D
speaktime	tell the time, E-D
undelete	undeletes a file, E-D
cnvapidhm	converts Apple][low, medium and high res pictures to IFF, E-D
menued	menu editor produces C code for menus, E-D
quick	quick disk-to-disk nibble copier, E-D
quickEA	copies Electronic Arts disks, removes protection, E-D
txed 1.3	demo of text editor from Microsmiths,E-D

C programs

spin3	rotating blocks graphics demo, S-E-D
popcli	start a new CLI at the press of a button, like Sidekick, S-E-D
vsprite	VSprite example code from Commodore, S-E-D
AmigaBBS	Amiga Basic bulletin board prog., S-D

Assembler programs

star10	makes star fields like Star Trek intro,S-E-D

Pictures

Mount Mandelbrot	3D view of Mandelbrot set
Star Destroyer	hi-res Star Wars starship
Robot	robot arm grabbing a cylinder

Texts

vendors	Amiga vendors, names, addresses
cardco	fixes to early Cardco memory boards
cinclude	cross-reference to C include files
mindwalker	clues to playing the game well
slideshow	make your own slideshows from the Kaleidoscope disk

AMICUS Disk 13

Amiga Basic programs

Routines from Carolyn Scheppner of CBM Tech Support, to read and display IFF pictures from Amiga Basic. With documentation. Also included is a program to do screen prints in Amiga Basic, and the newest BMAP files, with a corrected ConvertFD program. With example pictures, and the SaveILBM screen capture program.

Routines to load and play FutureSound and IFF sound files from Amiga Basic, by John Foust for Applied Visions. With documentation and C and assembler source for writing your own libraries, and interfacing C to assembler in libraries. With example sound.

Executable programs

gravity	Sci Amer Jan 86 gravitation graphic simulation, S-E-D

Texts

MIDI	make your own MIDI instrument interface, with documentation and a hi-res schematic picture.

AMICUS Disk 14

Several programs from Amazing Computing issues:

Tools

Dan Kary's	C structure index program, S-E-D

Amiga Basic programs:

BMAP Reader	by Tim Jones
IFFBrush2BOB	by Mike Swinger
AutoRequester	example
DOSHelper	Windowed help system for CLI commands, S-E-D
PETrans	translates PET ASCII files to ASCII files, S-E-D
C Squared	Graphics program from Scientific American, Sept 86, S-E-D
crlf	adds or removes carriage returns from files, S-E-D
dpdecode	decrypts Deluxe Paint, remo
ves copy	protection, E-D
queryWB	asks Yes or No from the user returns exit code, S-E
vc	VisiCalc type spreadsheet, no mouse control, E-D
view	views text files with window and
slider	gadget, E-D
Oing, Sproing, ya	Boing, Zoing are sprite-based
Boing! style	demos, S-E-D
CLIClock, sClock, wClock	are window border clocks, S-E-D

Texts

An article on long-persistance phospor monitors, tips on making brushes of odd shapes in Deluxe Paint, and recommendations on icon interfaces from Commodore-Amiga.

AMICUS Disk 15

The C programs include:

'pr'	a file printing utility, which can print files in the background, and with line numbers and control character filtering.
'fm'	displays a chart of the blocks allocated on a disk.
'Ask'	questions an 'execute' file, returns an error code to control the execution in that batch file
'Stat'	an enhanced version of AmigaDOS 'status' command.
'Dissolve'	random-dot dissolve demo displays IFF picture slowly, dot by dot, in a random fashion.
'PopCLI2'	invoke new CLI window at the press of a key.

The executable programs include:

'Form'	file formatting program through the printer driver to select print styles
'DiskCat'	catalogs disks, maintains, sorts,merges lists of disk files
'PSound'	SunRize Industries' sampled sound editor & recorder
'iconmaker'	makes icons for most programs
'Fractals'	draws great fractal seascapes and mountain scapes.
'3D Breakout'	3D glasses, create breakout in a new dimension
'AmigaMonitor'	displays lists of open files,
memory use,	tasks, devices and ports in use.
'Cosmoroids'	version of 'asteroids' for the Amiga.
'Sizzlers'	high resolution graphics demo written in Modula 2.

Texts:

'ansi.txt'	explains escape sequences the CON: device responds to.
'FKey'	includes template for making paper to sit in the tray at the top of the Amiga keyboard.
'Spawn'	programmer's document from Commodore

Amiga, describs ways to use the Amiga's multitasking capabilities in your own programs.

AmigaBasic programs:

'Grids'	draw sound waveforms, and hear them played.
'Light'	a version of the Tron light-cycle video game.
'MigaSol'	a game of solitaire.
'Stats'	program to calculate batting averages
'Money'	"try to grab all the bags of money that you can."

AMICUS 15 also includes two beautiful IFF pictures, of the enemy walkers from the ice planet in Star Wars, and a picture of a cheetah.

AMICUS Disk 16

'juggler' demo by Eric Graham. a robot juggler bouncing three mirrored balls, with sound effects. Twenty-four frames of HAM animation are flipped quickly to produce this image. You control the speed of the juggling. The author's documentation hints that this program might someday be available as a product.

IFF pictures

parodies of the covers of Amiga World and Amazing Computing magazines.

C programs:

'inputhandler'	example of making an input handler.
'FileZap3'	binary file editing program
'ShowPrint'	displays IFF picture, and prints it.
'Gen'	program indexes and retrieves C structures and variables declared in the Amiga include file system.

Executable Programs:

'FixHunk2'	repairs an executable program file for expanded memory
'ms2smus'	converts Music Studio files to IFF standard 'SMUS' format. I have heard this program might have a few bugs, especially in regards to very long songs, but it works in most cases.
'Missile'	Amiga version of the 'Missile Command' video game,

This disk also contains several files of scenarios for Amiga Flight Simulator II. By putting one of these seven files on a blank disk, and inserting it in the drive after performing a special command in this game, a number of interesting locations are preset into the Flight Simulator program. For example, one scenario places your plane on Alcatraz, while another puts you in Central Park

AMICUS Disk 17

Telcommunications disk whuch contains six terminal programs.

"Comm" V1.33	term prog. with Xmodem, WXmodem,
"ATerm" V7.2	term prog. includes Super Kermit
"VT-100"V2.6	Dave Wecker's VT-100 emulator with Xmodem,Kermit, and scripting
"Amiga Kermit"	V4D(060) port of the Unix C-Kermit
"VTek" V2.3.1	Tektronix graphics terminal emulator based on the VT-100 prog. V2.3 and contains latest 'arc' file compression
"AmigaHost"	V0.9 for Compuserve. Includes RLE graphics abilities & CIS-B file transfer protocol.
"FixHunk"	expansion memory necessity
"FixObj"	removes garbage characters from modem received files
"Txt"	filters text files from other systems to be read by the Amiga E.C.
"addmem"	executeable version for use with mem expansion article in AC v2..1
'arc'	file documentation and a basic tutorial on un 'arc'ing files
"arcrx"	for making "arc" files E.C.

AMICUS Disk 18

Logo	Amiga version of the populacomputer language, with example programs, E-D
Tv"Text	Demo version of the TV"Text character generator
PageSetter	Freely distributable versions of the updated PagePrint and PageIFF programs for the PageSetter desktop publishing package.
FullWindow	Resizes any CLI window using only CLI commands, E-D
Life3d	3-D version of Conway's LIFE program, E-D
Defdisk	CLI utility to re-assign a new Workbench disk, S-E-D
Calendar.WKS	Lotus-compatible worksheet that makes calendars
SetKey	Demo of keyboard key re-programmer, with IFF picture to make function key labels, E-D
VPG	Video pattern generator for aligning monitors, E-D
HP-10C	Hewlett-Packard-like calculator, E-D
SetPrefs	Change the Preferences settings on the fly, in C, S-E-D
StarProbe	Program studies stellar evolution. C source included for Amiga and MS-DOS, S-E-D
ROT	C version of Colin French's AmigaBasic ROT program from Amazing Computing. ROT edits and displays polygons to create three dimensional objects. Up to 24 frames of animation can be created and displayed. E-D
Scat	Like lng, windows on screen run away from the mouse, E-D
DK	Decays" the CLI window into dust, in Modula 2, S-E-D
DropShadow2	Adds layered shadows to Workbench windows, E-D

AMICUS Disk 19

This disk carries several programs from Amazing Computing. The IFF pictures on this disk include the Amiga Wake part T-shirt logo, a sixteen-color hi-res image of Andy Griffith, and five Amiga Live! pictures from the Amazing Stories episode that featured the Amiga.

Solve	Linear equation solver in assembly language, S-E-D
Gadgets	Bryan Catley's AmigaBasic tutorial,
Household	Bryan Catley's AmigaBasic household inventory program, S-D
Waveform	Jim Shields' Waveform WorkAmigaBasic, S-D
DiskLib	John Kennan's AmigaBasic disk librarian program, S-D
Subscripts	Ivan Smith's AmigaBasic subscript example, S-D
String, Boolean	C programs and executables for Harriet Maybeck Tolly's Intuition tutorials, S-E-D
Skinny C	Bob Riemersma's example for making small C programs, S-E-D
COMAL.h	Make C look like COMAL header file,
EmacsKey	Makes Emacs function key definitions by Greg Douglas, S-D
AMon 1.1	Snoop on system resource use, E-D
BTE	Bard's Tale character editor, E-D
Size	CLI program shows the size of a given set of files, E-D
WinSize	CLI window utility resizes current window, S-E-D

AMICUS Disk 20

Compactor, Decoder Steve Michel AmigaBasic tools, S-D

BobEd	BOB and sprite editor written in C,S-E-D
SpriteMasterII	Sprite editor and animator by Brad Kiefer, E-D
BlitLab	Blitter chip exploration C program by Tomas Rokicki, S-E-D
FPic	Image processing program by Bob Bush loads and saves IFF images, changes them with several techniques, E-D
Bankn	Complete home banking program, balance your checkbook! E-D

AMICUS Disk 21

Target	Makes each mouse click sound like a gunshot, S-E-D
Sand	Simple game of sand that follows the mouse pointer, E-D
PropGadget	Harriet Maybeck Tolly's proportional gadget example, S-E
EHB	Checks to see if you have extra-half-bright graphics, S-E-D
Piano	Simple piano sound program
CelScripts	Makes cel animation scripts for Aegis Animator, in AmigaBasic

This disk has electronic catalogs for AMICUS disks 1 to 20 and Fish disks 1 to 80. They are viewed with the DiskCat program, included here.

AMICUS Disk 22

Cycles	Light cycle game, E-D
Show_PrintII	Views and prints IFF pictures, including larger than screen
PrtDrvGen2.3	Latest version of a printer driver generator
Animations	VideoScape animations of planes and boing ball
Garden	Makes fractal gardenscapes
BasicSorts	Examples of binary search and insertion sort in AmigaBasic

AMICUS Disk 23

An AMICUS disk completely dedicated to music on the Amiga. This disk contains two music players, songs, instruments, and players to bring the thrill of playing "Big Sound" on your Amiga

Instruments	a collection of 25 instruments for playing and creating music. The collection ranges from Cannon to Marimba
List INSTR	program to list the instruments DMCS will not load as well as list the origins for any instrument.
Music	a collection of 14 Classical pieces
1812Overture	The 16 minute classical feature complete with Cannon!

Three Amiga Music Players:

SMUSPlay
MusicCraft2SMUS
MusicStudio2SMUS

AMICUS Disk 24

Sectorama	A disk sector editor for any AmigaDOS file-structured device, recover files from a trashed hard disk. By David Joiner of MicroIllusions
Iconize	Reduces the size of IFF images, companion program, Recolor, remaps the palette colors of one picture to use the palette colors of another. Using these programs and a tool to convert IFF brushes to Workbench icons, make icons look like miniatures of the pictures.
CodeDemo	Modula-2 program converts assembler object files to inline CODE statements. Comes with a screen scrolling example
AmiBug	Workbench hack makes the same fly walk across the screen at random intervals. Otherwise, completely harmless.
BNTools	Three examples of assembly language code from Bryce Nesbitt: 1. SetLace.prog to switch interlace on&off. 2. Why, replace AmigaDOS CLI Why 3. Loadlt, prog to load a file into memory until a reboot. (Only the most esoteric hackers will find Loadlt useful.)
Monolace	CLI program resets Preferences to several colors of monochrome & interlace screens. C source is included, works with DisplayPref, a CLI program which displays the current Preferences settings.
BoingMachine	A ray-traced animation of a perpetual motion Boing-making machine, includes the latest version of the Movie program, which has the ability to play sounds along with the animation. By Ken Offer
Daisy	Example of using the translator and narrator devices to make the Amiga talk. It is written in C.
QuickFlix	Script-driven animation and slideshow program flips through IFF images.
BMon	System monitor AmigaBasic program ; perform simple manipulations of memory.
Moose	Random background program, a small window opens with a moose resembleing Bullwinklesaying witty phrases user definable.
DGCS	Deluxe Grocery Construction Set, simple Intuition-based prog for assembling and printing a grocery list.
The Virus Check	directory holds several programs relating to the software virus that came to the US from pirates in Europe as detailed in Amazing Computing V2.12. Bill Koester's full explanation of the virus code is included. One program checks for the software virus on a Workbench disk; the second program checks for the virus in memory, which could infect other disks.

AMICUS Disk 25

Nemesis	Graphics demo pans through space towards the mythical dark twin of the sun with wonderful music and space graphics.
The KickPlay	directory holds text that describes several patches to the Kickstart disk. For Amiga 1000 hackers who feel comfortable patching a disk in hexadecimal, KickPlay offers the chance to automatically do an ADDMEM for old expansion memory, as well as the ability to change the picture of the "Insert Workbench" hand. A program is also included for restoring the correct checksum of the Kickstart disk.
KeyBird	BASIC prog edits keymaps, adjust the Workbench keymaps or create your own.

8ColorWB	Modifies the Workbench so three bitplanes are used, icons can have eight colors, instead of four, eight-color icons are included. Public domain program "zapicon" or "brush2icon" converts eight-color IFF brushes to icons, to use Deluxe Paint to make icons for this new Workbench.
BrushIcon	Converts brushes to icons (bizarr docs).
Egraph	Graphing prog reads [x,y] values from a file and displays them on the screen, similar to the same-named Unix program.
Keep 1.1	Message-managing program for telecommunications, lets you save messages from an online transcript to another file, understands the message format of the national networks and several types of bulletin board software. Moves through the transcript and save messages
Kill.fastdir	Speed up directory access, it creates a small file in each directory on a disk which contains the information about the files, will also remove all the "fastdir" files from each directory. by CLImate's authors
The LaceWB program changes between interlace and non-interlace Workbench. Previously, you were forced to reboot after changing Preferences to an interlaced screen. This program flips between the normal and extended screen heights.	
PW_Utility	A shareware utility for ProWrite users, changes margin settings and font types.
Guru	A CLI program, prints out probable causes for Guru mediations; C source included.
DiskWipe	Latest from Software Distillery, removes files from directories or disk drives, much faster than "delete."
Snow	AmigaBasic makes snowflake designs.
Mlist	Mailing list database.
Softballstats	Maintain softball statistics/ team records.
Dodge	Short Modula-2 program moves the Workbench screen around after a period of time, prevents monitor burn-in.

AMICUS Disk 26

Todor Fay's SoundScape module code from his Amazing Computing articles. The source to Echo, Chord, TX, and VU is included. The Lattice and Manx C source code is here, along with the executable modules.	
ImageMaker	Interesting tool edits Image structures for C, loads & saves C code directly.
Ciaz2	Update of prog to convert IFF images to PostScript files for printing on laser printers
SDBackup	Hard disk backup prog with Lempel-Ziv compression to reduce the necessary number of disks.
TCB	Prints information about tasks and processes in the system; assembler source is included.
FunBut	Lets a function key act like a rapid series of left mouse button events.
DC	A handy program for people who use an Amiga 1020 5 1/4 inch drive as an AmigaDOS floppy. A Workbench program that sends a DiskChange signal to the operating system: instead of typing "diskchange df2:" over and over again, just click on the icon. C source included.
System config	File makes screen 80 columns wide of text in the Scribble! word processor.
Dick2Ram	2 programs to move the Scribble! spelling dictionary to and from the RAM disk.
Lexical	Analyzes a text file and gives the Gunning-Fog, Flesch, and Kincaid indices which measure readability.
HexDump	Modula-2 program to display memory locations in hexadecimal.
Tartan	AmigaBasic; design Tartan plaids.
DirMaster	Disk catalog program.
BMP	plays 8SVX sampled sounds in the background while something else is happening in the Amiga, as your Amiga is booting, for example.
ShowPt	CLI program changes your pointer to a given pointer.
AMICUS 26 also has a collection of mouse pointers, & Workbench program to display them	

Fred Fish Public Domain Software

Fred Fish Disk 1:

amigademo	Graphical benchmark for comparing amigas.
amigaterm	simple communications program with Xmodem
balls	simulation of the "kinetic thingy" with balls on strings
colorful	Shows off use of hold-and-modify mode.
dhrystone	Dhrystone benchmark program.
dotty	Source to the "dotty window" demo on the Workbench disk.
freedraw	A small "paint" type program with lines, boxes, etc.
gad	John Draper's Gadget tutorial program
gfxmem	Graphical memory usage display prog.
halfbrite	demonstrates "Extra-Half-Brite" mode, if you have it
hello	simple window demo
lattfp	accessing the Motorola Fast Floating Point library from C
palette	Sample prog. to design color palettes.
trackdisk	Demonstrates use of the trackdisk driver.
requesters	John Draper's requester tutorial and example program.
speech	Sample speech demo program. Stripped down "speechtoy".
speechtoy	Another speech demo program.

Fred Fish Disk 2:

alib	Object module librarian.
cc	Unix-like frontend for Lattice C compiler.
dbug	Macro based C debugging package. Machine independent.
make	Subset of Unix make command.
make2	Another make subset command.
microemacs	Small version of emacs editor, with macros, no extensions
portar	Portable file archiver.
xrf	DECUS C cross reference utility.

Fred Fish Disk 3:

gothic	Gothic font banner printer.
roff	A "roff" type text formatter.
tf	A very fast text formatter
cforth	A highly portable forth implementation. Lots of goodies.
xlisp	Xlisp 1.4, not working correctly.

Fred Fish Disk 4:

banner	Prints horizontal banner
bgrep	A Boyer-Moore grep-like utility
bison	CNU Unix replacement 'yacc', not working.
bm	Another Boyer-Moore grep-like utility
grep	DECUS grep
kermit	simple portable Kermit with no connect mode.
MyCLI	Replacement CLI for the Amiga. V. 1.0
mandel	A Mandelbrot set program, by Robert French and RJ Mical

Fred Fish Disk 5

cons	Console device demo program with supporting macro routines.
freemap	Creates a visual diagram of free memory
input.dev	sample input handler, t raps key or mouse events
joystick	Shows how to set up the gameport device as a joystick.
keyboard	demonstrates direct communications with the keyboard.
layers	Shows use of the layers library
mandelbrot	IFF Mandelbrot program
mouse	hooks up mouse to right joystick port
one.window	console window demo
parallel	Demonstrates access to the parallel port.
printer	opening and using the printer, does a screen dump, not working
print.support	Printer support routines, not working.
proctest	sample process creation code, not working
region	demos split drawing regions
samplefont	sample font with info on creating your own
serial	Demos the serial port
singlePlayfield	Creates 320 x 200 playfield
speechtoy	latest version of cute speech demo
speech.demo	simplified version of speechtoy, with IO requests
text.demo	displays available fonts
timer	demos timer.device use
trackdisk	demos trakcdisk driver

Fred Fish Disk 6:

compress	like Unix compress, a file squeezer
dadc	analog clock impersonator
microemacs	upgraded version of microemacs from disk 2
mult	removes multiple occuring lines in files
scales	demos using sound and audio functions
setparallel	Allows changing parallel port parameters
setserial	Allows changing serial port parameters.
sortc	quicksort based sort program, in C
stripc	Strips comments and extra whitespace from C source

Fred Fish Disk 7:

This disk contains the executables of the game Hack V 1.0.1.

Fred Fish Disk 8:

This disk contains the C source to Hack on disk 7.

Fred Fish Disk 9:

moire	Draws moire patterns in black and white
MVP-FORTH	Mountain View Press Forth, version 1.00.03A. A shareware version of FORTH from Fantasia Systems.
proff	a more powerful text formatting program
setface	Prog to toggle interlace mode on and off
skewb	a rubic's cube type demo
sparks	moving snake Graphics demo

Fred Fish Disk 10:

conquest	An interstellar adventure simulation game
dehex	convert a hex file to binary
filezap	Patch program for any type of file.
fixobj	Strip garbage off Xmodem transferred files.
iff	Routines to read and write iff format files.
ld	simple directory program
ls	Minimal UNIX ls, with Unix-style wildcarding, in C
sq,usq	file squeeze and unsqueeze
trek73	Star Trek game
yachtc	Dice game.

Fred Fish Disk 11:

dpslide	slide show program for displaying IFF images with miscellaneous pictures

Fred Fish Disk 12:

amiga3d sign".	Shows a rotating 3 dimensional solid "Amiga
ArgoTerm	a terminal emulator program, written in assembler
arrow3d	Shows a rotating 3 dimensional wire frame arrow.
ld4	directory listing program
IconExec	
SetWindow	two progs for launching progs from Workbench, presently only works under CLI.
SetAlternate	Makes an icon show a second image when clicked once
StarTerm	terminal emulator, with ASCII Xmodem, dialer, more.

Fred Fish Disk 13:

A Bundle of Basic programs, including:

Jpad	toybox	ezspeak	mandelbrot
xmodem	3dsolids	addbook	algebra
ror	amgseq1	amiga-copy	band
bounce	box	brickout	canvas
cardfi	circle	colorcircles	Copy
cubes1	cutpaste	datedogstar	
dragon	draw	dynamictriangle	
Eliza	eterm	fillibuster	fractal
fscape	gomoku	dart	haiku
hal9000	halley	hauntedM	hidden
join	loz	mandel	menu
minipaint	mouse	Orthello	patch
pena	pinwheel	gboxrandom-circles	
Readme	rgb	rgbtest	Rord
sabotage	salestalk	shades	shapes
shuttle			
sketchpad	spaceart	speakspeach	
speecheasy	spell	sphere	
spiral	striper	superpad	suprshr
talk	terminal		
termtest	tom	topography	triangle
wheels	xenos	xmostriper	

(note: some programs are Abasic, most are Amigabasic, and some programs are presented in both languages)

Fred Fish Disk 14:

amiga3d	update of #12, includes C source to a full hidden surface removal and 3D graphics
beep	Source for a function that generates a beep sound
dex	extracts text from within C source files
dimensions	demonstrates N dimensional graphics
filezap	update of disk 10, a file patch utility
gfxmem	update of disk 1, graphic memory usage indicator
gi	converts IFF brush files to Image struct, in C text.
pdterm	simple ANSI VT100 terminal emulator, in 80 x 25 screen
shell	simple Unix 'csh' style shell
termcap	mostly Unix compatible 'termcap' implementation.

Fred Fish Disk 15:

Blobs	graphics demo, like Unix 'worms'
Clock	simple digital clock program for the title bar
Dazzle	An eight-fold symmetry dazzler program. Really pretty!
Fish	double buffered sequence cycle animation of a fish
Monopoly	A really nice monopoly game written in AbasiC.
OkidataDump	Okidata ML92 driver and WorkBench screen dump program.
Polydraw	A drawing program written in AbasiC.
Polyfractals	A fractal program written in AbasiC.

Fred Fish Disk 16:

A complete copy of the latest developer IFF disk

Fred Fish Disk 17:

The NewTek Digi-View video digitizer HAM demo disk

Fred Fish Disk 18:

AmigaDisplay	dumb terminal program with bell, selectable fonts
Ash	Prereelease C Shell-like shell program, history, loops, etc.
Browser	wanders a file tree, displays files, all with the mouse
MC68010	docs on upgrading your Amiga to use a MC68010
Multidim	rotate an N dimensional cube with a joystick
PigLatin	SAY command that talks in Pig Latin
Scrimper	Screen image printer
Xlisp1.6	source, docs, and execut for a Lisp interpret.

Fred Fish Disk 19:

BlackJack	text-oriented blackjack game
JayMinerSlides	Slides by Jay Miner, Amiga graphics chip designer, showing flowchart of the Amiga internals, in 640 x 400.
Keymap_Test	test program to test the key mapping routines
LockMon	Find unclosed file locks, for programs that don't clean up.

Fred Fish Disk 20:

AmigaToAtari	converts Amiga object code to Atari form
DiskSalv	program to recover files from a trashed AmigaDOS disk.
Hash	example of the AmigaDOS disk hashing function
Hd	Hex dump utility ala Computer Language magazine, April 86
MandelBrots	Mandelbrot contest winners
MultiTasking	Tutorial and examples for Exec level multitasking
Pack	strips whitespace from C source
PortHandler	sample Port-Handler program that performs. Shows BCPL environment
Random	Random number generator in assembly, f or C or assembler.
SetMouse2	sets the mouse port to right or left
SpeechTerm	terminal Emulator with speech capabilities, XModem
TxEd	Demo editor from Micrsmith's Charlie Heath

Fred Fish Disk 21

This is a copy of Thomas Wilcox's Mandelbrot Set Explorer disk. Very good!

Fred Fish Disk 22

This disk contains two new "strains" of microemacs.

Lemacs	version 3.6 by Daniel Lawrence. For Unix V7, BSD 4.2, Amiga, MS-DOS, VMS. Uses Amiga function keys, status line, execute, startup files, more.
Pemacs	By Andy Poggio. New features include <ALT> keys as Meta keys, mouse support, higher priority, backup files, word wrap, function keys.

Fred Fish Disk 23

Disk of source for MicroEmacs, several versions for most popular operating systems on micros and mainframes. For people who want to port MicroEmacs to their favorite machine.

Fred Fish Disk 24:

Conques	interstailer adventure simulation game
Csh	update to shell on Disk 14, with built in commands,named variables substitution.
Modula-2	A pre-release version of the single pass Modula-2 compiler originally developed for MacIntosh at ETHZ. This code was transmitted to the AMIGA and is executed on the AMIGA with a special loader. Binary only.

Fred Fish Disk 25

Graphic Hack	A graphic version of the game on disks 7 and 8 This is the graphics-oriented Hack game by John Toebes. Only the executable is present.

Fred Fish Disk 26

UniHunk	Processes the Amiga "hunk" loadfiles. Collect code, data, and bss hunks together, allows individual specificatio of code, data, and bss origins, and generates binary file with format reminiscent of Unix "a.out" format. The output file can be easily processed by a separate program to produce Motorola "S-records" suitable for downloading to PROM programmer. By Eric Black.
C-kermit	Port of the Kermit file transfer program and server.
Ps	Display and set process priorities
Archx	Yet another program for bundling up text files and mailing or posting them as a single file unit.

Fred Fish Disk 27

ABdemos	Amiga Basic demos; Caroly Scheppner.
NewConvertFD	creates .bmaps from fd files.
BitPlanes	finds addresses of and writes to bitplanes of the screen's bitmap.
AboutBmaps	A tutorial on creation and use of bmaps.
LoadILBM	loads and displays IFF ILBM pics.
LoadACBM	loads and displays ACBM pics.
ScreenPrint	creates a demo screen and dumps it to a graphic printer.
Disassem	Simple 68000 disassembler. Reads standard Amiga object files and disassembles the code sections. Data sections are dumped in hex. The actual disassembler routines are set up to be callable from a user prog so instructions in memory can be disassembled dynamically. By Bill Rogers.
DvorakKeymap	Example of a keymap structure for the Dvorak keyboard layout. Untested but included because assembly examples are few and far between. By Robert Burns
Hypocycloids	Spirograph, from Feb. 84 Byte.
LinesDemo	Example of proportional gadgets to scroll a SuperBitMap.
MemExpansion	Schematics and directions for building your own homebrew 1 Mb memory expansion, by Michael Fellinger.
SafeMalloc	Program to debug 'malloc()' calls
ScienceDemos	Convert Julian to solar and sidereal time, stellar positions and radial velocity epoch calculations and Galilean satellite plotter. By David Eagle.

Fred Fish Disk 28

ABasic games by David Addison: Backgammon , Cribbage, Milestone, and Othello	
Cpp	DECUS 'cpp' C preprocessor, & a modified 'cc' that knows about the 'cpp', for Manx C.
Shar	Unix-compatible shell archiver, for packing files for travel.
SuperBitMap	Example of using a ScrollLayer, syncing SuperBitMaps for printing, and creating dummy RastPorts.

Fred Fish Disk 29

AegisDraw Demo	Demo program without save and no docs.
Animator Demo	Player for the Aegis Animator files
Cc	Unix-like front-end for Manx C
Enough	Tests for existence of system resources, files, and devices
Rubik	Animated Rubik's cube program
StringLib	
Vt100	VT-100 terminal emulator with Kermit and Xmodem protocols

Fred Fish Disk 30

Several shareware programs. The authors request a donation if you find their program useful, so they can write more software.

BBS	an Amiga Basic BBS by Ewan Grantham
FineArt	Amiga art
FontEditor	edit fonts, by Tim Robinson
MenuEditor	Create menus, save them as C source, by David Pehrson
StarTerm3.0	Very nice telecom by J. Nangano

(Fred Fish Disk#30 is free if requested when ordered with at least three other disks from the collection.)

Fred Fish Disk 31

Life	Life game, uses blitter to do 19.8 generations a second.
Mandelbrot	Version 3.0 of Robert French's program.
MxExample	Mutual exclusion gadget example.
RamSpeed	Measure relative RAM speed, chip and fast.
Set	Replacement for the Manx "set" command for environment variables,with improvements.
Tree	Draws a recursive tree, green leafy type, not files.
TxEd	Crippled demo version of Microsmith's text editor, TxEd.
VDraw	Full-featured drawing program by Stephen Vermeulen.
Xicon	Invokes CLI scripts from icon
Ticon	Displays text files from an icon.

For PDS orders, please use form on page 112

Fred Fish Disk 32

Address	Extended address book, AmigaBASIC
Calendar	Calendar/diary program, AmigaBASIC
DosPlus1	First volume of CLI oriented developer tools
DosPlus2	2nd volume of CLI oriented developer tools.
Executables only:	
MacView	Views MacPaint pics in Amiga low or high res, no sample pictures, by Scott Evernden.
Puzzle	Simulation of puzzle with moving squares.
ShowHAM	View HAM pictures from CLI.
Solitaire	ABasiC games of Canfield and Klondike, from David Addison.
Spin3	Graphics demo of spinning cubes, double-buffered example.
Sword	Sword of Fallen Angel text adventure game written in Amiga Basic.
Trails	Leaves a trail behind mouse, in Modula-2

Fred Fish Disk 33

3dstars	3d version of the "stars" program below.
Bigmap	Low-level graphics example scrolls bitmap with ScrollVPort.
Dbuf.gels	Double-buffered animation example for BOBs and VSprites.
DiskMapper	Displays sector allocation of floppy disks.
MemView	View memory in real time, move with joystick.
Oing	Bouncing balls demo
Sproing	Oing, with sound effects.
ScreenDump	Dumps highest screen or window to the printer.
Sdb	Simple database program from a DECUS tape.
Stars	Star field demo, like Star Trek.
TermPlus	Terminal program with capture, library, function keys, Xmodem, CIS-B protocols.
Vt100	Version 2.0 of Dave Wecker's VT-100 emulator, with scripts & function

Fred Fish Disk 34

Alint	Support files for Gimpel's 'lint' syntax checker
Blink	PD 'alink' compaible linker,faster,better.
Browser	Updated to FF 18 'browser', in Manx, with scroll bars, bug fixes.
Btree	b-tree data structure examples
Btree2	Another version of 'btree'
Calendar	Appointment calendar with alarm.
Less	File viewer, searching, position by percent, line number.
NewFonts	Set of 28 new Amiga fonts from Bill Fischer
Pr	Background print utility, style options, wildcards.
Requester	Deluxe Paint-type file requester, with sample.

Fred Fish Disk 35

ASendPacket	C example of making asynchronous I/O calls to a DOS handler, written by C-A
ConsoleWindow	C example of getting the Intuition pointer a CON: or RAW: window, for 1.2, by C-A.
DirUtil	Walk the directory tree, do CLI operations from menus
DirUtil2	Another variant of DirutIl.
FileRequester	Lattice C file requester module, with demo driver, from Charlie Heath.
MacView	Views MacPaint pictures in Amiga low or high res, with sample pictures, by Scott Evernden.
Plop	Simple IFF reader program
PopCLI	Sidekick-style program invokes a new CLI, with automatic screen blanking.
QuickCopy	Devenport disk copiers duplicate copy-protected disks.
ScrollPf	Dual playfield example, from C-A, shows 400 x 300 x 2 bit plane playfield on a 320 x 200 x 2 plane deep playfield.
SendPacket	General purpose subroutine to send AmigaDos packets.
SpriteMaker	Sprite editor, can save work as C data structure. Shareware by Ray Larson.
Tracker	Converts any disk into files, for electronic transmission. Preserves entire file structure. Shareware by Brad Wilson.
TriClops 3-D	space invasion game, formerly commercial, now public domain. From Geodesic Publications.
Tsize	Print total size of all files in subdirectories.
Unifdef	C preprocessor to remove given #ifdef'd sections of a file, leaving the rest alone. By Dave Yost
Vttest	VT-100 emulation test program. Requires a Unix system.

Fred Fish Disk 36

Acp	Unix-like 'cp' copy program
Clock	Updated version of clock on disk 15.
Csh	Manx 'csh'-like CLI, history, variables, etc.
DietAid	Diet planning aid organizes recipes, calories
Echo	Improved 'echo' command with color, cursor addressing
FixHunk	Fixs programs to let them run in external memory.
Fm	Maps the sectors a file uses on the disk.
KickBench	Docs, program to make a single disk that works like a Kickstart and Workbench.
Lex	Computes Fog, Flesch, and Kincaid readability of text files.
TunnelVision	David Addison ABasic 3D maze perspective game.
Vc	Visicalc-like spreadsheet calculator program.
Vt100	Version 2.2 of Dave Wecker's telecom program
YaBoing	Oing! style game program shows sprite collision detects

Fred Fish Disk 37

This disk is a port of Timothy Budd's Little Smalltalk system, done by Bill Kinnersley at Washington State University.

Fred Fish Disk 38

CSquared	Sep 86 Sci American, Circle Squared algorithm
FixObj	Strips garbage off Xmodem transfered object files
Handler	AmigaDOS handler (device) example from C-A
Hp-10c	Mimics a HP-10C calculator, written in Modula-2
IFFEncode	Saves the screen as an IFF file
IffDump	Dumps info about an IFF file
Jsh	BDS C-like CLI shell
NewStat	STATUS-like program, shows priority, processes
Reversi	Game of Reversi, version 6.1
UUdecode	Translate binary files to text, Unix-like programs
Vdraw	Drawing program, version 1.14
VoiceFiler	DX MIDI synthesizer voice filer program
Window	Example of creating a D OS window on a custom screen

Fred Fish Disk 39

AnsiEcho	'echo', 'touch', 'list', 'cls' written in assembler.
Display	Displays HAM images from a ray-tracing program, with example pictures.
Driver	Example device driver source, acts like RAM: disk
Xlisp	XLisp 1.7, executable only

Fred Fish Disk 40

Ahost	Terminal emulator with Xmodem, Kermit and CIS B protocols, function keys, scripts, RLE graphics and conference mode.
AmigaMonitor	Dynamically displays the machine state, such as open files, active tasks, resources, device states, interrupts, libraries, ports, etc.
Arc	Popular file compression system, the standard for transiting files
AreaCode	Program that decodes area codes into state and locality.
Blink	'alink' replacement linker, version 6.5
Cosmo	An 'asteriods' clone.
Dg210	Data General D-210 Terminal emulator
DirUtil	Windowed DOS interface program, V 1.4
DOSHelper	Windowed AmigaDOS CLI help program
PagePrint	Prints text files with headers, page breaks, line numbers
PopCLI	Starts a new CLI with a single keystroke, from any program, With a screen-saver feature. Version 2, w&ource.
SpriteEd	Sprite Editor edits two sprites at a time
X-Spell	Spelling checker allows edits to files

Fred Fish Disk 41

AmigaVenture	Create your own text adventure programs in AmigaBasic.
Csh	Version 2.03 of Dillon's C sh-like shell. Executable only
Dbug	Macro based C debugging packageupdate to FF #2
DualPlayField	example from CBM, update to Intuition manual
GetFile	Heath's file requester, with source
LatXref	Cross reference of Lattice 3.10 header files
Lines	Line drawing demo program
SetFont	Changes font used in a CLI window
Vt100	Version 2.3 of the VT-100 terminal program.

Fred Fish Disk 42

This disk contains an Amiga version of MicroGNUEmacs.

Fred Fish Disk 43

BasicBoing	AmigaBasic program demos page flipping of a 3D cube
Bbm	Demo copy of B.E.S.T. Business Management System.
BbsList	A list of Amiga Bulletin Board Systems
Cc	C compiler frontends for Manx and Lattice C
Copper	A hardware copper list disassembler
InstIFF	Converts Instruments demo sounds to IFF sampled sounds
PopColours	Adjust RGB colors of any screen
SpriteClock	Simple clock is displayed on a sprite above all screens
ST Emulator	Non-serious Atari ST emulator
WBrun	Lets Workbench programs be run from the CLI
Wild	Two Unix shell style wild card matching routines

Fred Fish Disk 44

Icons	Miscellaneous icons
NewIFF	New IFF material from CBM for sampled voice and music files
RayTracePics	The famous ray-tracing pictures, from FF#39, now converted to IFF HAM format for *much* faster viewing.
ViewILBM	Displays normal and HAM ILBM files

Fred Fish Disk 45

Clue	Clue board game
Make	Another 'make', with more features
Pictures	Miscellaneous pictures
Update	Updates older disk with newer files fromanother disk
WhereIs	Searches a disk for files of given name

Fred Fish Disk 46

Asm	Shareware 68010 macro assembler, ROM Kernal Manual compatible
CheckModem	'execute' file program detects presence of modem
Egad	Gadget editor from the Programmers Network
Jive	Transforms a file from English to Jive.
MyJib	A binary only copy of Ma tt's alternate runtime library. Author: Matt Dillon
ProffMacros	Subset Berkeley 'ms' and 'mm' macros for 'proff'
ValSpeak	Transforms a file from English to Valley Speak.

Fred Fish Disk 47

3D-Arm	Simulation of a robotic a rm, very good graphics, teaching tool, including C source.
Juggler	Eric Graham's stunning HAM animation of a robot juggler
VT-100	Version 2.4 of Dave Wecker's terminal emulator, with Xmodem and Kermit file transfer protocols

Fred Fish Disk 48

Bru	Alpha version of a hard disk file archiver
Comm	Version 1.30 of a terminal emulator with phone directories
Csh	Version 2.04 of Matt Dillon's Unix 'csh'-like CLI replacement, including Lattice & Manx C source
Diskperf	Disk benchmark program for Unix and Amiga
Du	Computes disk storage of a file or directory
MemWatch	Program to watch for programs that trash low memory. It attempts to repair the damage, and puts up a requester to inform you of the damage. From the Software Distillery.
Profiler	A realtime execution profiler for Manx C programs. Includes C source.

Fred Fish Disk 49

Cycloids	Update of electronic spirograph from disk 27
DirUtil	Enhanced version of DirUtil from disk 35
MultiDef	Scans a set of object modules and libraries searching for multiply defined symbols
MyUpdate	Disk update utility with options for stripping comments from C header files, and interactive verification of the updating process
Plot	Computes and displays 3 dimensional functions in hires
Polygon	Moire type pattern generator with color cycling
QMouse	Queries whether a mouse button is pressed. This can give a return code that can customize a startup-sequence based on whether a mouse button was pressed.
Touch	Exampie of setting the datestamp on a file, using a technique from Commodore-Amiga
Trees	More extensive version of the trees program on Disk 31

Fred Fish Disk 50

Asm	Version 1.1 of a shareware 68000 macro assembler, compatible with the Metacomco assembler. This includes an example startup module and more Motorola mneumonics.
BreakOut	A brick breakout game, uses 3-D glasses
DiskZap	Version 1.1 of a program to edit disks and binary files
FirstSilicon	A smart CLI replacement with full editing and recall of previous commands
Missile	A Missile Command-type game, with sound, in assembler
PerfectSound	Sound editor for a low-cost sound digitizer
Sizzlers	Graphics demos
UnixArc	Ver of 'arc' for Unix System V machines,in C
Wombat	Version 3.01 of Dave Warker's terminal emulator

Fred Fish Disk 51

Bison	GNU for Unix 'yacc', working update to FF4
Compress	Update to the file compression program on Disk 6
Cos	"Wheel of Fortune"-type game in AmigaBasic
DiffSsed	Unix-like 'dif' and 'ssed' for finding the differences between two files, and then recreating the other, given one file, and the list of differences.
Sq, Usq	Portable versions of the CP/M squeeze and unsqueeze

Fred Fish Disk 52

Assign	Replacement for AmigaDOS 'assign' command in C
Fractal	Makes random fractal terrains
Poly, HAMPoly	Workbench-type demos for making polygons in lores and HAM
MxGads	Example of mutual exclusion gadgets with GadgetText
Tek4010	Tektronix 4010 terminal emulator
VDraw	Versions 1.16 and 1.19 of a Deluxe Paint-like drawing program

Fred Fish Disk 53

Animations	Demo animations with player program for Aegis Animator
ARCre	Creates rename scripts for files with long names, so they can be easily 'arc'ed and un'arc'ed.
ARP	Preliminary AmigaDOS replacements for 'break', 'cd', 'chmod', 'echo', 'filenote' and 'makedir'
Compiler	Not fully ported to the Amiga, this is a 68000 C compiler. It wil produce simple assembly language output, but needs a lot of work.
Spreadsheet	Update with source of the 'vc' spreadsheet on disk 36
TarSplit	Port of program to split Unix 'tar'archives
UUencode	Utilities to encode and decode binary files for ASCII transmission, expanding them by 35%

Fred Fish Disk 54

Hanoi	Solves Towers of Hanoi Problem in it's own Workbench window. by Ali Ozer
ISpell	Port of a Unix screen oriented, interactive spelling checker. (Expansion RAM required) by Pace Willisson
Ing	A Screen of lots of bouncing little windows by Leo 'Bols Ewhac' Schwab
Lav	Displays number of tasks in run queue, averaged over last 1, 5, and 15 minute periods. by Willaim Rucklidge
MIDITools	Programs to play/record through the MIDI I/F. by Fred Cassirer
MoreRows	Program to make the Work Bench Screen larger than normal. by Neil Katin and Jim Mackraz
Tilt	Program to make your Amiga look like it didn't pass vibration testing. by Leo ' Bols Ewhac' Schwab

Fred Fish Disk 55

Csh	V2.05 of Matt Dillon's csh like shell (Modified for Manx C). by Matt Dillon, Modified by Steve Drew
NewStartups	New C Startup modules:
AStartup.asm	with 1.2 fixes and better quote handling.
TWStartup.asm	opens a stdio window, using user specs. by Commodore, posted to BIX by Carolyn Schepper
Palette	Change another program's screen colors. by Carolyn Schepper
PipeDevice	Allows the standard output of one process to be fed to the standard input of another. by Matt Dillon
ScreenSave	Save a normal or HAM mode screen as an IFF file. by Carolyn Schepper
ShanghaiDemo	Demo of the Activision game Shanghai.
SoundExample	A double buffered sound example for Manx C. by Jim Goodnow
Vsprites	A working vsprite example. by Eric Cotton
Vt100	V2.6 of Dave's Vt100 terminal emulator with kermit and xmodem. by Dave Wecker

Fred Fish Disk 56

ClipBoard	Clipboard device interface routines, to provide a standard interface. by Andy Finkle
ConPackets	Demos the use of DOS Packets, ConUnit, etc. by Carolyn Schepper
GetDisks	Program to find all available disk device names and return them as an exec list. by Philip Lindsay
GetVolume	Program to get volume name of the volume that a given file resides on. by Chuck McManis
Icon2C	Reads an icon file and writes out a fragment of C code with the icon data structures. by Carolyn Schepper
MergeMem	Program to merge the MemList entries of sequentially configured RAM boards. by Carolyn Schepper
mCAD	An object oriented drawing program, V1.1 by Tim Mooney

Fred Fish Disk 57

Replaced by FF97 Due to Copyright problems

Fred Fish Disk 58

ASDG-rrd	Extremely useful shareware recoverable ram disk. by Perry Kivolowitz
BigView	Displays any IFF picture, independent of the physical display size, using hardware scroll. by John Hodgson
EGraph	Reads pairs of x and y value from a list of files and draws a formatted graph. by Laurenece Turner
HyperBase	Shareware data management system. V1.5
MemClear	Walks through the free memory lists, zeroing free memory along the way. by John Hodgson
NewZAP	A third-generation multi-purpose file sector editing utility. V3.0 by John Hodgeson
RainBow	A Maurauder-Style rainbow generator. by John Hodgson
SMUSPlayers	Two SMUS plays, to play SMUS IFF music formatted files. by John Hodgson
View	A tiny ILBM viewer by John Hodgson
WBdump	JX-80 optimized workbench printer that does not use DumpRPort. by J. Hodgson

Fed Fish Disk 59

Browser	Update to browser program on disks 18 and 34. S-E
Browser2	Another different browser program. E
Clock	Clock program with fonts, colors. E
Dme	Dillon text editor V1.22 for programmers,ED
DropCloth	Puts pattern on Workbench backdrop,ED
DropShadow	Puts shadows on Workbench windows,E-D
FixWB	Similar to DropCloth, but doesn't work yet. S-D
mCAD	Object-oriented drawing program, version 1.2.2. Much improved over disk 56.
Robotroff	Demo of animated pointers on Workbench. S-E-D
Supermort	General compounding/amortization loan calculator. E-D

Fred Fish Disk 60

Various shareware and freeware programs

Blitz	Memory resident file viewer. Very fast. E-D
BlitzFonts	Makes text output faster. E-D
HandShake	Terminal emulator with VT52/VT100/ VT102support. E-D
Med	Mouse-driven text editor version 2.1. E-D
PrtDrvGen	Generates printer drivers, version 1.1.S available from author. E-D
Show	Slideshow-like IFF viewer, V2.1. E-D
Uedit	Customizable text editor V2.0. E-D
Ueturbo	Example Uedit setup macros. S-E-D

Fred Fish Disk 61

ATPatch	Patches Transformer to work under AmigaDOS 1.2. S-E-D
FillDisk	Writes zeroes to free blocks on a disk for security. S-E-D
LPatch	Patch for programs that abort when loading under AmigaDOS 1.2. S-E-D
MicroEmacs	Conroy MicroEmacs V3.8b, newer than disk 22. S-E-D
PearlFont	Like Topaz, but rounded edges.
Terrain	Generates fractal scenery. S-E-D
VSprites	Makes 28 Vsprites, from P&E Book.

Fred Fish Disk 62

This is a port of the Unix game 'Hack', by the Software Distillery, version 1.0.3D.

Fred Fish Disk 63

This is a port of the Unix game 'Larn', by the Software Distillery, version 12.0B.

Fred Fish Disk 64

This is an offical IFF specification disk from Commodore, an update to disk 16.

Fred Fish Disk 65

Bawk	Unix text processor, like 'awk'. Doesn't work, but source is included. S-E-D.
MWB	Example of rerouting Workbench window open calls to another custom screen. Version 1.01, S-E-D
CloseWB	Example for closing a custom Workbench screen. S-E-D
Cookie	Generates one-line fortune-cookie aphorisms. S-E-D
JTime	Build-your-own mouse port clock.
MenuBuilder	Creates C source files for menus, based on text descriptions. S-E-D.
NewPackets	CBM tutorial on new packets and structures in AmigaDos 1.2.
PascalToC	Pascal to C translator, not so great. S-E-D
Prep	'ratfor'-like FORTRAN preprocessor. S-E-D
RunBack	Starts programs from CLI, allowing CLI window to close. E-D

For PDS orders, please use form on page 112

SunMouse This program automatically clicks in windows when the mouse is moved over them. V1.0, E-D

Fred Fish Disk 66

AmScsi Preliminary plans for a SCSI disk controller board.
Asm68k Macro assembler, version 1.0.1. E-D
Assigned Example for avoiding DOS insert-disk requester, by scanning the list of 'assign'ed names. S-E-D
Dk Pretends to eat away at CLI window. S-E-D
Flip Flips whole screen as a joke. S-E-D
Foogol Foogol cross-compiler generates VAX assembly code. S-E-D
Free Prints amount of free space on all drives.S-E-D
MallocTest malloc/free memory test program. S-E-D
Melt Pretends to melt the screen. S-E-D
Nart Graphic flying string demo. S-E-D
Purty Easy way to set printer attributes from Workbench. E-D
RayTracer Simple ray tracing program. E-D
SendPackets Updated CBM examples of packet routines on disk 35. S-E-D
SnapShot Memory resident screen dump. E-D
TagBBS Shareware BBS system, version 1.02.

Fred Fish Disk 67

AmCat Shareware disk cataloging program.
AmigaSpell Shareware intuition spelling checker, V2.0. E-D
Bouncer 3-D bouncing ball written in MultiForth, SED
Comm Terminal program version 1.33, E
Dux5 Another version of DirUtil. S-E-D
HexCalc Hex, octal, & decimal calculator. E-D
Icons Various big and alternate image icons.
Mandala Mandala graphics and sound. E
PersMail Demo shareware personal file manager.
RSLClock Menu bar clock version 1.3. E-D
RTCubes Graphics demo of 3D cubes. E-D
Wheel "Wheel of Fortune"-type game,AmigaBASIC

Fred Fish Disk 68

This is version MG 1b of the MicroGNUEmacs. Source and executable areincluded, as well as source for other computers besides the Amiga.

Fred Fish Disk 69

Asm68k Macro assembler, v1.0.3, E-D
BlitLab Blitter exploring program, in C, S-E-D
Conman Replacement console device handler adds editing and history to any application that uses CON:, v0.9, E-D
Console Replacement console routines, in C, S-E-D
Dk Decays the screen bit by bit, update to disk 66, in Modula-2, S-E-D
Frags Displays memory fragmentationby listing the size of free memory blocks, in C, S-E-D
IconType Change the type of an icon, in C, S-E-D
Make 'make' in Manx C, S-E-D
MonProc Monitors processes for packet activity, in C,S-ED
MouseClock Mouse pointer into a digital clock,in C,SED
Sb Browses system structures, from Transactor magazine, v1.0, in C, S-E-D
Spew Generates 'National Enquirer'-type headlines from rules file. In C,S-E-D
Spool Three programs to demonstrate multitasking & spooling in a printer spooler. In C, v1.2, S-E-D
Wc Counts words ala Unix 'wc', but faster, in C,S-ED

Fred Fish Disk 70

This is a disk of shareware programs.

AmigaMonitor Explores state of the system, v1.13
Arc Standard file compressor andl librarian, v0.23, a port of MS-DOS v5.0. E-D
BlackBook Phone book program.
DoTil Intuition-driven file manipulator program,v2.0.
GravityWars Game of planets, ships and black holes,v1.03.
Jobs Alternate user interface to CLI and WB, v2.1.
Lens Magnifies area around mouse, shows it in a window, v1.0.
Life-3d 3D version of the classic cellular-automaton game, v1.2.
Logo Logo language interpreter
SetKey Demo keymap editor, v1.0
Vpg Makes displays for aligning video monitors, v1.0.

Fred Fish Disk 71

AirFoil Makes airfoils using the Joukowski transformation, in C, S-E-D
Amiga Basic Miscellaneous programs including 3D plot program, a kaleidoscope, C-A logo drawing program file comparison utility string search program, S-E-D
Blocks A variation of "lines", but with variable color blocks. E-D
Comm Great terminal program, v1.34, E-D
DiskX Utility for exploring file system.E-D
Fpic Simple image processing program that operates on IFF pictures, with several filters, merging images, E-D
IconMk Makes icons for files , v1.2a, E-D
Icons New icons
NewFonts Two new fonts; 'shall18', an electronic ciruit element font, and 'ibm5', a PC-like font.
PetCLI An AmigaBASIC CLI shell program.
PWDemo Demo of the commercial product PowerWindows,v1.2. It aids creation of custom windows, menus, and gadgets, giving C or assembly source. E-D
Rot Creates and animates 3-D objects, v0.5, E,D
TimeSet Sets time from Workbench, E-D

Fred Fish Disk 72

This is a disk of IFF pictures.

Fred Fish Disk 73

Add Customizes existing program menus with Amiga-key shortcuts. Also includes 'until', which waits until a given window is created. Shareware, in C, S-E-D.

AutoIconOpen Fools WB into thinking mouse has double-clicked icons. In C, S-E-D
Dio Generic Exec device interface code for opening libraries, getting multiple I/O channels, asynchronous operations, etc. In C, S-E-D.
Dissolve Slowly displays IFF files, ala Nov 86 Dr. Dobb's program. In C, S-E-D
DTerm Flexible, reprogrammable terminal program v1.10, E-D
Expose Re-arranges windows so that at least one pixel of menu bar gadgets are exposed. InC, S-E-D.
Lit Scans a text file, converts to C-style printable strings.C,v2.0, S-E-D
Lmv "Long Movie", program views series of IFF picts in quick succession, upto 19 fps. Shareware, E-D
MouseOff Mouse pointer disappears after ten seconds of non-use. In C, S-E-D
ParOut Examples of controlling parallel port with resources instead of the PAR: device.In C,S-E-D
PenPalFont Child-like font.
RunBackGround Similar to RunBack on disk 66, runs program from the CLI allowing the CLI window to close. In C,S-E-D
SnapShot Screendump utility,update FF 66.E-D
TypeAndTell Example installs a device handler before Intuition, and speaks each key as it is pressed. In C and assembler, S-E-D
Xpior Prints info about system lists, in assembler,S-E-D

Fred Fish Disk 74

Cled Edits and recalls CLI commands, v1.3, E-D
Control Intercepts graphic printer dump calls and accesses color map, width,and screen resolution. C,S-E-D
Dme Simple WYSIWYG text editor for programmers,v1.25. Update of FF 59.E-D
DropShadow WB dropshadows, v2.0.UpdateFF59. E-D
Funds AmigaBASIC prog tracks mutual or stocks-D
Less Text viewing program, like Unix "more", v1.1, update to disk 34. S-E-D
Makemake Scans C source files and constructs a vanilla 'makefile' in the current directory. S-E-D
mCAD Object-oriented drawing prog. v1.2.4, update to FF 59.Shareware, E-D
Random Simple random number generator in C. S-E-D
TDebug Monitors devices by intercepting Exec SendIO() and DoIO() vectors, in C, v1.0, S-E-D
Units Converts measurements in different units, includes "chart" option, in C, S-E-D
XCopy Replacement for AmigaDOS 'copy', doesn't change the date, uses Unix wildcards. E-D

Fred Fish Disk 75

Bezier Play with Bezier curves points and granularity, S-E-D
BSplines Play with b-splines, as above, S-E-D
Comm C source for Comm terminal program v1.34. S-E-D
Copy Replacement 'copy' command v1.0, preserves date, in C, S-E-D
Diff Simple 'diff' in C, S-E-D
DuM2 Another DirUtil in Modula-2, v1.5, S-E-D
Eless Fast 'dir' program in C, S-E-D
Fd Faster 'eless' in C, S-E-D
HardCopy Sends a transcript of a CLI session to a file, in C, S-E-D
MouseOff Update FF73, turns off mouse pointer, S-E-D
SetFont Changes the font in a Workbench screen, v2.0, S-E-D
SpeedDir Another fast 'dir', in assembler, S-E-D

Fred Fish Disk 76 & 77

These are disks 1 and 2 of Chris Gray's Draco distribution for the Amiga. Draco is a compiled, structured language reminiscent of both C and Pascal. A full interface to AmigaDOS and Intuition is supplied. Be sure to get both disk 76 and 77.

Fred Fish Disk 78

Cycles Cycle game like "Tron", v1.0, E-D
EOMS Experts Only Mercenary Simulator game, E-D
MandelVroom Mandelbrot generator with enhanced palette controls, fixed/floating point, presets, v1.50, in Manx C, S-E-D

Fred Fish Disk 79

AsmTools CLI tools in assembler: echo, loadit, mounted, setlace, why; S-E-D
AssignDev Give devices multiple names, in C, S-E-D
AuxHandler Example of a dos handler that allows use of a CLI via the serial port. Includes source. Author: Steve Drew
Cmd Redirects printer output to a file, in C, S-E-D
Info AmigaDOS 'info' replacement, in C and assembler, S-E-D
Kill Removes a task and its resources, in C,S-E-D
M2Error Displays errors from TDI Modula-2 compiles, S-E-D
MoniProc Update to process packet prog. from FF69,in C,S,E,D
Mounted Program for testing if a drive is present, in a script In C, S-E-D
Nro Another 'roff'-style text formatter, in C, S-E-D
ParTask Finds parent task, in C, S-E-D
QueryAny For scripts, asks a question, accepts Y/N, gives return code. In assembler, S-E-D
ScnSizer Resets pref settings for screen size, in C,SED
SharedLib Example, shared lib, in C & assembler, S-E-D
Task Simple CreateTask() example in C, S-E-D
Uw Unix Windows client v1.0, in C, S-E-D
Who Lists tasks on ready and wait queues, in C, S-E-D

Fred Fish Disk 80 (see Fred Fish 90)

Fred Fish 80 has been withdrawn due to copyright problems.

Fred Fish Disk 81

Asm68k V1.1.0 of a macro assembler
AutoFacc Shrinks the FACC window and moves it to the back
Brushes 53 custom IFF brushes of electronic symbols
CheckIFF Checks structure of an IFF file CledV1.4 update FF74 of a simple CLI
Conman Replaces console handler to add editing and history to many programs
Fonts Miscellaneous fonts
Icon V6.0 of the Icon programming language
KeyLock Freezes the keyboard and mouse until pass word entered.

ScatDisplay hack created from "Ing"
Smush Smushes an IFF file.
Target Each mouse clikc becomes a gunshot

Fred Fish Disk 82

Adventure Port of the classic Crowther and Woods game
AmicTerm V0.50 of a telecommunications program, with scripts, redial, beeps, enhanced file requester
D2D-Demo Demo version of Disk-2-Disk by Central Coast Software
DX-Synth Voice filer program for Yamaha DX series synthesizers, update to disk 38
DiskMan V1.0 of another DirUtil program
Icons Miscellaneous new icons
Panl Universal MIDI patch panel, v1.2
Rocket Another Workbench hack, plays Lunar Lander
Sand Game of sands following your pointer.

Fred Fish Disk 83

This disk contains a demo version of TeX from N Squared. It is limited to small files, and the previewer can only display ten pages or less, and only a small number of fonts are provided.

Fred Fish Disk 84

AudioToolsPrograms from Rob Peck's July/August Amiga World article
BlitLab Blitter experimentation program, V1.2, update to FF69
Ed Simple editor, similar to Unix 'ed', based on the editor in Software Tools.
GravityWars Game of planets, ships and black holes, v1.04, update to disk 70.
HunkPad Adds legal padding to executables for Xmodem transmission.
PipeHandler An AmigaDOS pipe device which supports named pipes and taps. V1.2
PopCLI V3.0 of a hot-key to invoke a CLI window, with screen blanker, update to disk 40.
Requester Update FF34, file requester similar to DPaint.
ScottDevice V33.1 of a 'mount'able MicroForge SCSI driver.
Viacom Another Schwab hack, makes TV-like static on screen.Parody

Fred Fish Disk 85

Csh V2.06 of Dillon's 'csh'-like shell
FileReq Source to wildcard file requester
Hide Hides expansion memory from programs
ImageTools Shareware tools to manipulation IFF images
LowMem ServerShared library to aid in low memory situations
Plot6 A star plotting program with source.
RawIO Example of setting raw mode on standard input
Rocket Lunary Lander for Workbench, with source.
VMore "more"-like text viewing utility, v1.0 SE
Vnews Simple Unix news reader.

Fred Fish Disk 86

AutoPointAuto-selects window under the mouse pointer, with screensaver.
ClickToFront Double-clicks in window brings it to front, v1.1, S-E-D
Cmd V3.0 of a tool to redirect printer output toa file.
FileIISG-Demo Demo of Softwood File IIsg, a database manager with sound and graphics.

Fred Fish Disk 87

AdvSys Adventure system from Byte May 1987, v1.2 E-D
AutoIconOpen Fools Workbench to open disk icons, V1.2 update to disk 73, S-E-D
Claz Converts IFF files to PostScript, V2.0, SED
Commodi tiesMackraz's Commodities Exchange, an exec library to manage input handler, v0.4
Diff Update to disk 75 of Unix-like 'diff', S-E-D
Dme V1.27 of Dillon's text editor, update FF74,E-D
DropShadow V2.0 of prog. that puts shadows on Workbench, S-E-D
Elib Shared library example in Manx C.
ID-Handler An AmigaDOS device handler generates unique identifiers, V1.0, S-E-D
Install Alternate AmigaDOS 'install' programs,SED
MemWatch Waits for low memory trashing, V2.0, SED
MovePointer Moves pointer to given location, S-E-D
MoveWindow Move window to given location, S-E-D
MunchingSq Munching Squares hack, S-E-D
PalTest Test to see if this is a PAL machine, S-E-D
Sc Generates random scenery, S-E-D
Tek4695 Tek4695 printer driver
WBDualPF Example of dual-playfield screen, update FF41, S-E-D
WarpText Fast text rendering routines, S-E-D
Yaiff rExample IFF reader, S-E-D
Zoo A file archiver like "arc", v1.42A, E-D

Fred Fish Disk 88 (see Fred Fish 89)

FF Disk 88 has been removed due to copyright problems

Fred Fish Disk 89 (replaces Fred Fish 80)

DirMaster Disk catalogue program, V1.0a, E-D
FuncKey Shareware function key editor, V1.01, E-D
MFF-Demo Demo of MicroFiche Filer database prog
ScreenShift Adjust screen position ie Preferences,SED
Snake Bouncing squiggly lines demo, S-E-D
AutoEnquirer screen contraption requester improvement S-E-D
DemoLition Display Hack S-E-D

Fred Fish Disk 90 (replaces Fred Fish 80)

AmiGazer Night sky viewer of 1573 stars, set date, time, day. E-D
CardFile AmigaBasic card file study aid. E-D
Conman Console handler replacement gives line editing and history to most progs, v0.98,ED
IMandelVroom Slight update to disk 78 Mandelbrot program, E-D
NewDemos Replacements for lines and boxes demos that take less CPU time, E-D
Othello Game of Othello, E-D
PrinText Displays text files with gadgets, speech, IFF display, v1.2, E-D
PrtDrvGen Automatic printer driv. generator,v2.2b,ED
RainBench Cycles colors of WB backdrop or text. ED
ShortCut Makes single-key shortcuts for entering commonly typed CLI commands,&custom macros. E-D
ShowPrint Displays and prints all sizes of IFF pictures & controls printer output styles, v2.0 E-D
Sizzlers Graphics demos, v1.7.0, E-D
Timer Small Workbench timer counts time and $/ minute, E-D

Fred Fish Disk 91

Adventure Definition Language (ADL) a superset of an older language called DDL by Michael Urban, Chris Kostanick, Michael Stein, Bruce Adler, and Warren Usui. ADL enhancements by Ross Cunniff. Included are sources to the ADL compiler, interpreter, and debugger. Binaries combined by Ross with Lattice 3.03. CLI enviornment only. Documentation is available from the authors.

Fred Fish Disk 92

As6502 portable 6502 assembler, C source, by J. Van Ornum, Amiga port by Joel Swank
Bawk Text processor update from FF65 Inspired by UNIX awk. Searches files for patterns, performs actions based on patterns. By Bob Brodt; Amiga port by Johan Widen
HunkPad update of FF84 version. by J. Hamilton,pads an object file to a multiple of 128 bytes for better xmodem transfer. S E
Less Like Unix "more", better, version 1.2 update of FF74. Scrolls Back and f orward. S E by Mark Nudelman, Amiga port by Bob Leivian.
Ndir Library that implements the 4BSD unix dir access routines by MikeMeyer. S
Parse Recursive descent expression parcer, computes, and prints expressions. includes transcendental function support. c Source included. by J Olsen
Shar Two programs to pack and unpack shell archives includes C source. by Fabbian G. Dufoe
SmallLib 8 times smaller Amiga.lib replacement, binary only. by Bryce Nesbitt
UUencode Encode/decode binary files for e mail or text-only methods. Update of FF53, includes checksum technique, compatible with older versions, plus transparent to older versions options. By Mark Horton, modified by Alan Rosenthal & Bryce Nesbitt.

Fred Fish Disk 93

Dme Version 1.27 WYSIWYG programmer editor. Not a word processor. Includes key mapping, fast scro-lling, title-line statistics, multiple windows, ability to iconify windows. Update of FF87, SE. by Matt Dillon
MicroEmacs Version 3.8, update to FF61 includes source. Orig by Dave Conroy modifications by Daniel Lawrence

Fred Fish Disk 94

AudioTools Demo programs from Rob Peck's July/Augustt issue of AmigaWorld on accessing the audio device. V2,update of FF84. S. by Rob Peck
ClickUpFront Similar in function to ClickToFront prog (FF86), bring windows tofront by clicking on any part of them. V 1.0. by Davide Cervone SE
HeliosMouse Automatically activate a window simply by moving the mouse pointer into the window. V 1.0. Includes source. By Davide Cervone
IFF2Ps Convert any IFF file to postscript for printing or viewing on a postscript compatible device. Version 1.2, by William Mason and Sam Paolucci E
ModulaTools Various Modula 2 prog. routines. by Jerry Mack
Terrain3d Pseudo-random 3d relief scenery generator, update of "sc", FF87. by Chris Gray, 3d by Howard Hull

Fred Fish Disk 95

Cmd redirects the serial.device or parallel. device output to a file. Capture print jobs, debug or "offline" printing.V4 By C Scheppner SE
CygnusEdDemo Demo of CygnusSoft's CygnusEd editor, a multipefile, multiple feature editor.Includes demo 3.0 of MandFXP, by CygnusSoft Software E
Gomf "Get Outa My Face" makes the Guru go away to allow clean-up & shutdown more cleanly. V 1.0, by Christian Johnsen E
Journal records sequence of mouse & keyboard events, stored in a file for future playback. Good for demos or documtenting bugs. E. by D. Cervone
MergeMem attempts merging of MemList entries of sequentially configured ram boards. Allows allocating a section of memory which spans both boards. V 2, update of FF56. by Carolyn Schepper SE
PrinterStealer Asimilar to "Cmd", allows diversion of output destined for printer to a file. Binary only, Source avai. from authors. by A. Livshits & J-M Forgeas
Record-Replay similar to "Journal", records and plays back mouse and keyboard events. B only. source avail. from authors, Alex Livshits & J-M Forgeas

Fred Fish Disk 96

AnimPlayer Animation reader and displayer by the combined efforts of Videoscape, Sculpt3D, Silver, Forms- in-Flight, and Animator:Appreticeby M Hashetal.
Chess Amiga port, non-Amiga interface. High playability. V 1.0, S. by J. Stanback, Amiga port by B. Leivian
Hackbench provides source for WB-Likeprog, for experimentation & validation of new interface ideas. Not a WB replacement. by Bill Kinnersley
Label Print labels with arbitray text. V 1.3, Source available from author, M.Hansen
LineDrawer Produces line drawings based on drawing commands stored in a text file. Includes demo that draws an outline map of the USA and state borders. V1.0, SE. by John Olsen
PopUpMenu Example code implementing pop-up menus, reasonably compatible with Intuition menus.SE. by Derek Zahn
Tek4695 Tektronix 4695/4696 printer driver. SE. by P Staub
TimeRam Fast and Chip ram test prog,E by B Takahashi
WarpText Fast text rendering routines, to be linked with application progs.Text display "as fast or faster than "blitz". V2.0 update of FF87, S by Bill Kelly

Fred Fish Disk 97

Replaces FF57 for Copywrite problems
CutAndPaste Implementaions of Unix cut and paste commands. by John Weald
Graphit Progran to plot simple functions in 2 or 3 dimensions. by Flynn Fishman
Juggler V1.2 of robot juggler animation. Uses HAM mode and ray tracing. by Eric Graham
MouseReader Shareware program to read text files & view IFF files using only the mouse. by William Betz
Splines Prog to demonstrate curve fitting & ren-dering techniques. by Helene (Lee) Taran
Shm Graphics demo, approximately simulates the motion of two interacting pendulums. Includes S by Chris Ediss

For PDS orders, please use form on page 112

Fred Fish Disk 98

Access 16 color terminal program based on Comm V1.,34. Includes Macro window, custom gadgets, colorized menus, etc. V. Beta 0.18 by Keith Young ,comm by D.J.James. E.

Backup Writes AmigaDos disks as the backup destin-ation. recover files from the backup disk. Requires manual decisions on disk structure. by Alan Kent SE

DCDemo DishCat 2.3, a disk catalog program, demo limited to cataloging 100 files at a time. by Ed Alford, MicroAce Software

HdDriver WD-1002-05 hard disk controller driver. Card capable of maintaining 3 hard disks and 4 floppies, the driver is capable of only one hard disk. by Alan Kent SED

QBase Quick-Base, a "MailBase Management utility", define and maintain a maximum of 200 records per file. by Kevin Harrise E

Thai Thai language quiz program. Speak or type english/Thai sentences from supplied file. by Alan Kent SE

Fred Fish Disk 99

A-Render Version .3 a Ray- Tracing Construction Set for the Amiga Computer by Brian Reed ED

Fred Fish Disk 100

Berserk Must see animation, by Leo Schwab

Conman Console handler replacement, provides line editing and command line histories transparent to application prog uses CON: windows. Shareware V1.0 by W Hawes. E.

WBLander Workbench display hack game, upgrade of "Rocket" on FF85, now with sound effects. By Peter da Silva. E

Fred Fish Disk 101

ClrPlane Circular plane generator for VideoScape3D. Generates a clockwise circular polygon with the specified number of vertibles. V1.0 by T Floryan SE

IconAssembler Change Workbench Icons with IFF-brush files by Stefan Lindahl E

Microspell Standalone spelling checker scans text files and reports errors. 1000 common word list, 43,000 word main dictionary with multiple user dictionary support. Interfaces with MicroEMACS 3.9 with an emacs macro to step through the source file, stopping at suspect words and allowing the user to option. V1.0 by Daniel Lawrence, SED

Midi midi library and utility set. Includes Midi monitor, routing utility, status utility, and more. by Bill Barton SED

PsIntrp Postscript Interpreter reads and previews files on screen. by Greg Lee S(assy)E

StartUps Three C startup file replacements for standard Astartup.obj and LStarup.obj. Options include (1) BothStartup.obj, for the WorkBench programs or CLI programs with or without command line parameters. (2) WBStartup.obj, for WorkBench programs or CLI programs that require no command line parameters. (3) CLIStartup.obj for CLI programs that require command line parameters but do not need to be WorkBench runnable. by Bryce Nesbitt SE

Fred Fish Disk 102

Dbug Machine independent macro based C de-bugging package. Update FF41. by F Fish profiling support by Binayak Banerjee SE

Match-stuff Heavy duty text pattern matching stuff. includes simple match text replacement capability. By Pete Goodeve

Sectorama Recover lost or damaged data from floppy or hard disks or repair a damaged volume. by David Joiner E

SliCon Smart input line interpreter with window for full editing. Upgrade FF50 by P Goodeve., E

Xicon Use Icons to call up scripts containing CLI commands. V2.0 upgrade of FF31.by Pete Goodeve E

Fred Fish Disk 103

AviTrees Library and test prog. implement routines for creating and using trees held in memory.S.

Calc A programmable RPN calculator.

Cref A C cross ref. prog. S.

DosKwik A pair of progs. allows you to save files to one or more floppies for quick loading. Doesn't store Dos format.

IntuiDos A prog. to improve control and handling of the material on all disks in "CLI-area" .

MFF-Update A text import util. for MicroFiche Filer (demo on FF 89) and updates to some PD disk library datatbases.

Pack-It Takes all files the files and dirs. on a disk & packs them into a single file. for modem.

Sol Amiga version of solitaire.

Fred Fish Disk 104

Analyticalc Is a large and powerful spreadsheet prog.

Fred Fish Disk 105

AsmProgs Misc. assembly tools. Includes some S.

BasicProgs LeastSquare solves least square probs ,graphs results,S.

Bison A replacement for unix "yacc" command. S.

Dmouse Another prog in the tradition of display hacks". S.

FlamKey Allows keyboard and mouse inputs to be locked until a password is entered.

GravityWars Game of planets,ships &black holes, v2.0 ,FF84 update.

IPo2C A util. to write a C-lang definition to mimic the intuition pointer.S

Pere-et-Fil Ex. of creating & using reentrant processes. S.

Record Replay Similiar to "Journal" v2.0 update to FF95.

Fred Fish Disk 106

Funckey Shareware function key editor, v1.1update to FF89. Source avail. from author(Anson Mah).

MoreArt A small selection of some Amiga artwork.

QuickFlix An IFF slideshow and cel animation prog,v0.13.

RistiNolla A Finnish game. Also called Go-Moku. v1.0

Fred Fish Disk 107

Csh V2.07 of Matt Dillon's csh like shell.S.

Diff A util.,similiar to other common "diff" programs.S.

ProSuite Provides ex. code of facilities such as FileIO Requester, XText, DoRequest. & tutorial on how to program the Amiga. Book 1.01.S

SVTools Some useful tools. S.

Fred Fish Disk 108

AList Dir listing prog. based on LD4 prg S

DirMaster Disk cataloger. v1.0b, update to FF89. S.

Dots-Perfect Printer Driver for an Epson MX80 printer with upgrade kit installed. S.

MonIDCMP Lets you monitor the IntuiMessages that pass through an IDCMP window. Prints the message class,mouse coordinates,qualifier values. Great for debugging. S.

PrintPop A util. to send common control settings to PRT: S.

Sectorama Utilities to recover lost or damaged data from floppies & hard disks. v1.1, an update toFF102.

Tek Vt100 emulator for a Tektronix 4010/4014. (V2.6) update to FF52. S.

Zoo File archiver, like "arc". v1.24B. update to FF87

Fred Fish Disk 109

Machine A new animation.

SimCPM A CP/M sim,simulates 8080 along with h19 emulationS.

UUpc Hook up your Amiga as a usenet node. S.

Fred Fish Disk 110

A68k A 68000 assembler written in C. S.

Pdc An optimizing C compiler for the 68000 processor. update to FF53. but not based on that code.

Fred Fish Disk 111

AmyLoad A graphical monitor of cpu, blitter, & memory use. Includes two components; load.device,monitors system parameters, & amyload, which is the user interface & display program. by Jeff Kelley SE

AssignDev Assigns multiple names to a given device. modified version of the original released on disk number 79. By Phillip Lindsay, mod by Olaf Seibert SE

Gauge Continuously displays memory usage in a vertical bar graph Binary only. By:Peter da Silva

HeliosMouse Another "sunmouse"prog. Automatically activates a window by mouse pointer V 1.1, update to FF94. By:Davide Cervone SE

Labels Alphabetic & numeric ordered cross reference lists of defined system constants. Recommended for debugging purposes only, use the symbolic values in progs! By:Olaf Seibert

Mandel Mandelbrot generator program, with bits & pieces of code from C. Heath & R.J. Mical. By:Olaf Seibert S

PopLife A PopCLI type that plays life all over your screen. Lots of bits & pieces from Tomas Rokicki's blitlab & John Toebes' PopCLI. By:Olaf Seibert S

Fred Fish Disk 112

BeachBirds Beach scene portrayed by sprites & sound 512K machine. By:Jerrold Tunnell B only.

Bully Pushes all open screens around (thus the name "bully"). Show more than one demo at a time By:Mike Meyer S

DropShadow Dropshadow V2.0, use with Bryce Nesbitt's Wavebench demo. B only. By:Jim Mackraz

HagenDemos "RGB" & "Focus". RGB requires one meg, B only. By:Joel Hagen

Viacom Latest version of viacom for use in conjuction with WaveBench demo. B only. By:Leo Schwab & Bryce Nesbitt

WaveBench A neat screen hack, & runs on 512K machines. For more laughs, try in conjunction with Viacom or Ds (Dropshadow). Includes S. By:Bryce Nesbitt

Fred Fish Disk 113

AmiCron Simple Unix "cron" type program,a background task uses a disk-resident table to automatically run certain tasks on a regular basis, at specific times. V 2.3, S. By:Steve Sampson, Amiga port by Rick Schaeffer

Dme V 1.28f of Matt's text editor. A simple WYSIWYG editor for programmers. Not a WYSIWYG word processor Features : arbitrary key mapping, fast scrolling, title-line statistics multiple windows, iconify windows,etc. Update to FF93, S. By:Matt Dillon

DosDev Example DOS device driver in Manx C. Version 1.10, includes S. By:Matt Dillon

M2Amiga Demo of M2Amiga. A fast single pass Modula-2 compiler with editor, linker, a small set of interface & standard libraries. Compiles only small demo programs by limiting codesize & imports. Further development of the ETHZ compiler on FF24. B only. Demos with Source. By:R. Degen, C. Nieder, M. Schaub, J. Straube (AMSoft)

NoIconPos Clears position info of any icons, allows WorkBench to pick a new place for the icon. Useful for disk & drawer icons where Snapshot rewrites the icon & the window information. Modula-2, another demo for M2Amiga By:Markus. Schaub

Fred Fish Disk 114

CDecl English to C (and vice versa) translator for C declarations, a must for anyone except possibly the most hardcore C guru. By:Graham Ross, S

Vt100 V2.7 of vt100 terminal emulator with kermit & xmodem file transfer. Includes a few bug fixes posted to Usenet shortly after the posting of v2.7. Update to FF55. Includes S. By:Dave Wecker

WBLander a special version of the WBLander program from FF100. Ending is unique. Effective use of sound, Includes S. By:Peter da Silva & Karl Lehenbauer

Fred Fish Disk 115

Killer Masterful Video commercial of the Amiga, Beatles music, requires one meg of memory to run. Binary only. By:R. Wilt

Marketroid Another devious sprite oriented demo with lots of "in" jokes. 512K required, includes S. By:Leo Schwab

Fred Fish Disk 116

Movies A ram animation system with three different example animations; Kahnankas, Rocker, & F-15. Kahnankas & Rocker run on a 512K Amiga & show off overscan HAM mode. Includes a animation player program (movie), animation builder programs (dilbm, pilbm), & a text/graphics display program (vilbm). By:Eric Graham & Ken Offer

Fred Fish Disk 117

AMUC_Demo A really neat horizontal scrolling demo that is a 2400 x 200 pixel 32 color IFF picture composed of digiview snapshots of members of the AMiga Users of Calgary, superimposed on a very wide picture of the Calgary Skyline. B only. By:Stephen Vermeulen & Stephen Jeans

ExP_Demo Demo version of Express Paint 1.1., used to create the scrolling demo picture in the AMUC_Demo drawer on thi disk. B only. By:Stephen Vermeulen

Fred Fish Disk 118

Empire Complete rewrite, in Draco, of Peter Langston's Empire . A multiplayer game of exploration, economics, war, etc, can last months.Use local keyboard or modem.V1.0, shareware, & S . By:Chris Gray, original game by Peter Langston

HAMmmm Displays lines whose end points are bouncing around the screen, which is a double buffered HAM screen. The Y positions of the points are continuously copied into an audio waveform and played on all four channels, & the pitch of a just intoned chord is derived from the average X position of these points. JForth. Source By:Phil Burk

Stars Based on original code by Leo Schwab, credits longer than actual demo. Runs on 512K Amiga. B only. By:Hobie Orris

WireDemo Demonstrates the Amiga's line drawing speed. Runs on a 512K Amiga. Includes S. By:Matt Dillon

Fred Fish Disk 119

MicroEMACS V3.9e of Daniel Lawrence's variant of Dave Conroy's microemacs. Update to FF93. Also included, for the first time, is extensive documentation in machine readable form. SE. Author: Dave Conroy, Enhanced by Daniel Lawrence

Fred Fish Disk 120

Amoeba Clone of Space Invaders, one of the PDS games for the Amiga. B only BY: LateNight Developments

BackGammon Graphical Backgammon (an undergraduate A.I. course project). Version 1.0, S. By: Robert Pfister

Bankn A complete checkbook system offered by the author as shareware. Version 1.3, binary only. By: Hal Carter

EgyptianRun "road race + hazards" type game. Version 1.1, B only, shareware, source available from author. By: Chris Hames

IconImage Replace an old icon image with a new image, without affecting icontype, drawer data, etc. SE. By: Denis Green

Fred Fish Disk 121

BasicStrip AmigaBASIC prog. helps to convert programs written in other forms of Basic to AmigaBASIC. By: George Trepal

DataPlot Shareware,AmigaBASIC, plotting program. Also includes a least squares curve fit program. By: Dale Holt

Plot Shareware 3-D AmigaBASIC graphing prog. & sample output plots. Source available via author.By: George Trepal

Stairs AmigaBASIC prog. demos a musical illusion based upon perceptual circularity of widely spaced tones whose volumes are defined as a sinusoidal relationship to their frequency. By: Gary Cuba

Uedit V2.3 of this nice shareware editor. With learn mode, command language, menu customization, and other user configurability and customizability features. Binary only, shareware, update toFF60. By: Rick Stiles

WBColors Prog to change Workbench colors for progs that expect to be booted off their distribution disk but are run from a hard disk. SE. Author: Stefan Lindahl

Fred Fish Disk 122

Asteriods Asteroid game. The images and sounds are replaceable by the end user. Anything goes! By: Rico Mariani

Iff2Pcs Interactive puzzle prog. takes any IFF file with up to 16 colors, and breaks it up into squares to make a puzzle which the user can then piece together. V1.0,S. By: Ali Ozer

Names A shareware program to create and manage mailing lists. Binary only. By: Ernie Nelson

Pr Utility to print listings in different formats. Similar to the Unix "pr" program. Includes source By: Samuel Paolucci

PushOver Board strategy game, AmigaBASIC. Push your pieces onto the board until yfive in a row in any direction.S. By: R.Yost

PuzzlePro Create a puzzle from an IFF picture, which the user can then piece back together again. AmigaBASIC. V1.0, B only, shareware, source available from author. By: Syd Bolton

Fred Fish Disk 123

Arp ARP stands for "AmigaDOS Replacement Project". Arp is an effort led by Charlie Heath of Microsmiths Inc., to replace the current DOS in a compatible fashion, so that current programs will continue to work. Arp also makes whatever improvements are possible, so that current and future programs will work better. Various authors contributed work

Car One of Allen's entries to the Badge Killer Demo Contest. It apparently is an inside joke relating to a well known Amigan's experience with a certain highend graphics hardware manufacturer. Author: Allen Hastings

Fred Fish Disk 124

Icons Some sample animated icons. By: L. Pfost

Tarot AmigaBASIC Nice graphic of tarot cards. Author: L.Pfost

Fred Fish Disk 125

ElGato Animation entry to the BK D Contest. Background music arrangement, requires Sonix to use. By Kevin Sullivans

Fred Fish Disk 126

Colour Manipulate the colors of specific named screens, saving current color sets to data files, loading new color sets from data files, or interactively changing colors. S. By J. Russell

Dance Two programs, "dancing polygons", are entries to the BKD Contest. They aresimilar, but demonstrate the range of colors available on the Amiga. S. By: John Olsen

HBHill Animation entry to the BDK Contest. First known animation using the "Extra Half Brite" mode. By: Kevin Sullivan

Iconify Subroutine creates an icon on the Amiga screen that can be subsequently dragged around, and double-clicked on. You can use this to have your programs "iconify" themselves to temporarily get out of the user's way. With source & demo program. By Leo Schwab

OnlyAmiga Animation entry to BDK Contest. Three balls being juggled by pyramids rotating on their tops. By Iqbal Singh Hans

Suplib Support library needed to rebuild various programs of Matt's from source, including DME, DTERM, etc. S By Matt Dillon

VCheck V1.2 of virus detection prog. from Commodore Amiga Technical Support.. Will test for the presence of a virus in memory, or on specific disks. B only. By: Bill Koester.

Fred Fish Disk 127

Bounce Entry for BDK Contest. Creates little dots that bounce around and multiply. S By: Steve Hansel and Tom Hansel

Nemesis Entry to BDK Contest. It is quite small for what it does, and won fifth place in the contest. B only. By Mark Riley

Ripples Entries to BDK Contest. Unlike most other animations, it shows a fixed object from a moving point of view, instead of vice versa. By: Allen Hastings

Fred Fish Disk 128

Dis 68000 disassembler, written in assembler. S By Greg Lee

DropCloth Place a pattern, a 2 bitplane IFF image or a combination of a pattern and image, into the WorkBench backdrop. Version 2.2, shareware, B By: Eric Lavitsky

LedClock An extremely simple clock program, for interlaced screens only. S By: Ali Ozer

MRBackUp Hard disk backup utility, does a file by file copy on AmigaDOS floppy disks. With an intuition interface & file compression. V1.3, Source. By: Mark Rinfret

Paint Simple screen painting program, written in web. Requires web preprocessing program to rebuild from source. Includes source in web. Author: Greg Lee

PrtDriver A printer driver for the Toshiba "3 in one" printer in its Qume (best) mode. Includes source in C and assembler. By: Rico Mariani

SDBackUp A hard disk backup utility. CLI interface only. Does file compression. V 1.1, binary only. By Steve Drew

Sed A clone of the Unix sed (Stream EDitor) program. Includes source. By: Eric Raymondw

Keys A "hot-keys" program binds keyboard function keys to window manipulation functions (window activation, front to back, moving screens, etc). S By: Davide Cervone

Fred Fish Disk 129

DosKwik A pair of programs which allow you to save files, or a group of files, to one or more floppies for quick loading, does not store files in DOS format, for speed. V2.0, update to FF103. B, Shareware. By: Gary Kemper

MRBackUp A hard disk backup utility, does a file by file copy to standard AmigaDOS floppy disks. Includes intuition interface & file compression. V2.0 (with sources) and 2.1 (binary only, source available from author). Update of FF128. By Mark Rinfret

PaintJet HP PaintJet printer driver from HP sources.

Patch Two independent ports of Unix utility "patch", which applies context diffs to text file to automatically update them. Patch V 1.3 was ported to the Amiga by Rick Coupland and patchV2.0 was ported by Johan Widen. S By: Larry Wall

Fred Fish Disk 130

DirMaster Shareware disk cataloger, V1.1, update of FF108, new features and enhancements. B only. By Greg Peters

Evo Human evolution toy/tutorial with source.By S. Bonner

Hp RPN calculator prog, supports calculations with binary, octal, decimal, hex, float, and complex numbers. Includes 32 registers for storing data & transcendental functions. V1.0, S By: Steve Bonner

Mach "mouse accelerator" prog. with hotkeys, features of sun mouse, clicktofront, and popcli, a title bar clock with a bbs online charge accumulator, etc.V1.6a, S By: Brian Moats

PatEdit A pattern editor for creating patterns to input to the Amiga SetAfPt macro call. Call sets the area fill pattern for the area filling graphics (RectFill, AreaDraw, etc). Includes source. By Don Hyde

QMan Mandelbrot generator written partially in assem. for speed. Includes source. By Steve Bonner

Fred Fish Disk 131

Dfc Copies disks like Maurader, but multitasks. Replaces diskcopy and format (smaller than either). Intuition interface. S By: Tomas Rokicki

HyperBase Shareware database management system. V1.6, Binary only, source available from authors. FF58 update. By: Michael MacKenzie, Marc Mengel, & Craig Norborg

Life A new version of Tomas's ancient Life game, with a new macro language for setting up patterns, good examples, S By: Tomas Rokicki

Mackie A Popcli replacement that draws pretty lines on the screen in blanking mode. Includes source. Author: Software Distillery; enhancements by Tomas Rokicki

Mg1b A version of Mg1b with an ARexx port and other improvements by Tomas Rokicki. Define macros & bind them to function keys in startup file. Includes source. Author: Various; enhancements by Rokicki

WFrags Another version of Frags, Pops up a little window that updates occasionally. Good for developers to monitor what progs. are doing to memory. S By: Tomas Rokicki

Fred Fish Disk 132

Berserk Animation, a "must see" for every Amiga user, and ranks with "Juggler" as a premier demo for the Amiga. The difference between this distribution, and FF100, is that includes "source". use it as an example for creating animations. Fred Fish felt it was appropriate to have at least one animation that was available at the "source code" level. Author: Leo Schwab

Fred Fish Disk 133

Conman Shareware replacement for the standard console handler, provides line editing and command line histories completely transparent to any application program that uses CON: windows. V1.1, binary only, updateoFF100. New features include additional editing keys, fast search keys, undo key, clear history command, and more. Author: William Hawes

Crc Two programs useful for generating 16-bit CRC listings of the contents of disks, and verifying that a given disk's files still compute to the same CRC's as listed. V1.0, binary only. By: Don Kindred

CrcLists Complete CRC check files for FF1-128 using the Crc program included on this disk. These were made directly from Fred's master disks. Author: Fred Fish

Overscan Patches the Intuition library so that sizable windows with MaxHeight of 200 (400 in interlace) and screens with Height of 200 (400 in interlace) will take advantage of the PAL overscan capability of Intuition V1.2. Useful only for European users who wish to run software written for the US market, without modifying the applications, but still using the additional space. S By Ari Freund

Fred Fish Disk 134

BoingThrows 50 frame HAM animation done with Sculpt-3D, and DigiPaint. The animation took about 325 hours of runtime to generate. By Marvin Landis

Browser Workbench tool, using text-only windows, makes all files in the system accessible for executing, copying, moving, renaming, deleting, etc. Billed as a "programmers workbench". V1.2, binary only. BY Peter da Silva

Dme V1.29 of Matt's text editor. Simple WYSIWYG editor designed for programmers. Arbitrary key mapping, fast scrolling, title-line statistics multiple windows, & ability to iconify windows. FF113 update,S. By Matt Dillon

For PDS orders, please use form on page 112

Find Utility searches for files that satisfy a given boolean expression of attributes, starting from a root pathname and searching recursively down through the hierarchy of the file system. Like the Unix find program. V1.0, includes source. By Rodney Lewis

Library Demo version of a shareware program that stores textual information without regard to structure or content, and allows complicated searching for specific patterns. , B only. By: Bill Brownson

SmartIcon Shareware Intuition objects iconifier. V1.0 is limited to iconifying windows, adds a new "iconify gadget" to each window, when clicked, iconifies the window into an icon in the ram: disk. B only, source available from author. By: Gauthier Groult

Fred Fish Disk 135

TeXF A selection of 78 TeX fonts, with a conversion program to convert them to Amiga fonts. 22 different fonts at various sizes, ranging from 15 pixels high to more than 150 pixels. Conversion program can also be used with the fonts distributed with AmigaTeX, yielding an additional 1000+ fonts for use withother Amiga programs. V2.5, binary only. By: Ali Ozer

Fred Fish Disk 136

AsmToolBox Assembler "toolbox" created to make interfacing between assembler programs and AmigaDOS easy. With source. By Warren Ring

Bison A replacement for unix "yacc" command. From the GNU (GNU is Not Unix) effort. Port of the latest GNU version, by William Loftus, with the goal of preserving all of bison's current features. Includes source & test pro. "calc". By: Bob Corbett and Rich& Stallman.

Iff2Pcs Interactive puzzle progr. takes any IFF file containing up to 16 colors, and breaks it into squares to make a puzzle the user can then piece back together again. V1.1, update of FF122, includes source. By Ali Ozer

Paste Version of the Unix paste utility. Paste concatenates corresponding lines of the specified files into a single output line (horizontal or parallel merging) or concatenates them into alternate lines (vertical or serial merging). S. By: David Ihnat

YaBoingII Game prog. demonstrating hardware sprite usage, including collision detection. Update of FF96. S. By: Ali Ozer, based on original by Leo Schwab

Zoo File archiver, like "arc" in concept, but different in implementation and user interface details. Includes features that "arc" lacks (such as file/path names up to 255 characters in length). V 1.71, update of FF108. B. By: Rahul Dhesi, port by Brian Waters

Fred Fish Disk 137

Ct Program to display images from a CT scanner, along with several interesting sample images of scans of real people, including a skull, brain, heart, and spine. Each image is 256 by 256 pixels in 2048 gray scale. The display software, a primitive user interface, is quite powerful, including functions like convolutions, averaging, laplacians, unsharp masking, edge detection, gradients, etc. Binary. by: J. Harman

JeansIcons Miscellaneous cute icons created for AMUC's monthly newsletter disk. Submitted by Stephen Vermeulen. Author: Steve Jeans

Muncho A cute little program which plays a digitized sound sample when you insert or remove a disk from your drive. If you don't like the sounds, you can replace them with your own. Binary only. By: Andrew Werth

Sit Update to the Set Icon Type prog. on FF107. V1.10, includes source. Author: Stephen Vermeulen

VGad A new gadget editor that takes two pictures of the window and its gadgets, one being the normal gadget state and the other being the fully selected state, then merges the data and converts to C source code. V1.0, binary only. Author: Stephen Vermeulen

VirusX A boot sector virus check program that runs in the background and automatically checks all inserted disks for a nonstandard boot sector. Such disks can optionally have their boot sector rewritten to remove the virus. Includes source. Author: Steve Tibbett

VLabel Program to print fancy customized disk labels. Combines an IFF picture and up to 50 lines of text (which may be placed arbitrarily in any font or point size) then print the result. The IFF picture can be virtually any size (up to 1008 by 1000). It will also print labels from a batch file produced by SuperBase. V1.20, binary only. By: Stephen Vermeulen

Fred Fish Disk 138

AmigaLine A series of various technical notes for Amiga programmers. By: Byrce Nesbitt

Diff Uses the same algorithm as the Unix diff program and also produces context diffs, suitable for use with patch. Binary only. By: Unknown (Decus C diff?)

Foreach A simple but useful program that expands a wild card file specification and then invokes the specified command once per expanded filename, with the expanded filename as the command argument. Includes source. Author: Jonas Flygare

MacFont A conversion tool to convert Mac fonts to Amiga fonts. Binary only. By: John O'Neill and Rico Mariani

ModulaTools Various useful routines for those using in Modula on the Amiga. Update to FF94, S . By: Jerry Mack

Vt100 Two new versions of Dave's vt100 terminal emulator. One version, based on vt100 2.6, has been enhanced by John Barshinger to include an iconify feature, add full 132 column support using overscan, and other features (binary only). The second version is release 2.8 of the main-stream version of vt100, as enhanced and supported by Tony Sumrall. S. By: Dave Wecker

Fred Fish Disk 139

AmiCron An enhanced and debugged version of AmiCron 2.3 from FF113. Includes source. By: Steve Sampson, Rich Schaeffer, Christian Balzer

ListScanner A nice little utility to display all the Exec lists. Simular to Xplor utility FF73. Includes source in assembler. By: Heiko Rath

ProCalc Simulates HP-11C programmable calculator. Both English & German versions. Shareware, B only. By: Gotz Muller

RemLib Removes a specified library (if currently unused) or displays some info on all available libraries. Source in assembler. By: Heiko Rath

TurboBackup A fast mass floppy disk duplicator with enforced verify mode to prevent errors. V1.0, binary only. By: Steffen Stempel and Martin Kopp

WArranger Sends a window, identified by its name, to the front or back, without selecting it. Useful with AmiCron. Works on all screens. Includes source in assembler. By: Heiko Rath

WheelChairSim A wheelchair simulator developed as a project for the Technical Resource Centre and the Albert Children's Hospital, to allow the matching of a wheelchair joystick to a child's handicap and allow the child to practice using the chair in a safe (simulated) environment. Binary only. Author: Unknown, submitted by Dr. Mike Smith

Fred Fish Disk 140

SBProlog Volume 1 of the 2 volume Stony Brook Prolog (SBP) distribution, V2.3.2. This volume contains the executables and libraries. Volume 2, on FF141, contains the C and Prolog Source. By: Logic Programming Group at SUNY, Stony Brook Amiga port by David Roch & Scott Evernden

Fred Fish Disk 141

SBProlog Volume 2 of the 2 volume Stony Brook Prolog (SBP) distribution, version 2.3.2. Volume 2 contains the C and Prolog source code. Volume 1, on FF140. By: Logic Programming Group at SUNY, Stony Brook Amiga port by David Roch and Scott Evernden

SmallC An Amiga port of the Small-C compiler, written by Ron Cain and published in Dr. Dobb's Journal, in about 1980. Small-C is a rather small subset of the full 'C' language. It is capable of compiling itself, and other small, useful programs. Requires an assembler and linker to complete the package and produce working executables. Source and binary. By: Ron Cain. Amiga port by Willi Kusche.

Fred Fish Disk 142

Diff Program uses same algorithm as Unix diff prog. and produces context diffs, suitable for use with patch.Same as FF138, but now includes the missing files (including source code). Author: Unknown (Decus C diff)

FracGen Generates fractal pictures from "seeds" you create. Unlike any of the other "fractal generators", it can be used to load and display previously created fractal pics, modify existing fractals, or create your own fractals. V1.1, B. By:D.Houck

SciSubr Scientific Subroutine Package from DECUS, ported to the Amiga to run with Absoft Fortran. A valuable resource of mathematical and statistical source code for those doing Fortran work on the Amiga. Author: Unknown; ported to the Amiga by Glenn Everhart

Fred Fish Disk 143

Rim RIM-5 (Relational Information Manager), a full relational DBMS suitable for VERY large databases using B-Tree data storage, crude (by today's standards) user interface, but full source code is provided. RIM runs on a wide variety of systems, small and large, and produce compatible databases. Includes a built in HELP database and a programming language. Full Fortran source & documentation included. by: Various, Amiga port by Glenn Everhart

Fred Fish Disk 144

AnalytiCalc V22-3D of Glenn Everhart's large and powerful spreadsheet program, update to FF104. Extra features "to have some pretentions of acting as an 'integrated system'. A virtual memory system supporting up to18000 columns and 18000 rows, multiple equations per cell, an outlining system, built-in cell annotation, and datafile access from any cell(s) of the sheet, plus an array of functions not present in most commercial spreadsheets. Source and documentation in arc'd form.

Fred Fish Disk 145

Csh Modification of csh like shell to provide file name completion and argument execution. Requires ARP 1.1. Binary only, but includes diffs for the reference 2.07 source base. Author: Matt Dillon; enhancements by Johan Widen

DMouse Versatile screen blanker, mouse blanker, auto window activator, mouse accelerator, popcli style programmable command key, pop window to front, push window to back, etc, widget. Very useful program! . V1.06, includes source. Author: Matt Dillon

Net Link protoco provides essentially an unlimited number of reliable connections between processes on two machines, where each can be either an Amiga or a Unix (BSD4.3) machine. Works on the Amiga with any EXEC device that looks like the serial.device. Works on UNIX with tty and socket devices. Achieves better than 95% average throughput on file transfers. V1.20, includes sources for both the Amiga and Unix versions. Author: Matt Dillon

Tab Tablature writing program, with intruments for a banjo and string guitar. Binary only. Author: Jeff deRienzo

TinyProlog VT-PROLOG is a simple prolog interpreter provided with full source code to encourage experimentation with the PROLOG language and implementations. Version 1.1, includes source. Author: Bill and Bev Thompson

Fred Fish Disk 146

Blanker2 A screen blanking program that turns the screen black after 90 seconds of keyboard and mouse inactivity. V1.27.88, includes source. Author: Joe Hitchens

C-Light A demo copy of a commercial ray tracing program, identical to commercial version but limited to ten objects per scene. Binary only. Author: Ronald Peterson

CrcLists Complete CRC check files for FF129-141 and FF143-145 of the library, using the crc program from FF133. Made directly from Fred's master library. FF142 omitted due to a problem with the crc program. by: Fred Fish

DmeMacros A set of DME macros which utilize templates to turn DME into a language-sensitive editor for C, Pascal, Modula-2, and Fortran. By Jerry Mack

MemoPad A shareware intuition-based memo reminder program. Nicely done. V1.1, binary only. by: Michael Griebling

Fred Fish Disk 147

MicroGNUEmacs MicroGNUEmacs(MG 2b) contains many additions and enhancements since the original works by Dave Conroy (credit belongs to all contributors and Beta testers. Note: Amiga specific source code files and the document files have been archived. An executable copy of the PDS archive program "Zoo" is in the "c" directory

Fred Fish Disk 148

EFJ "Escape from Jovi" A machine-code game featuring hi-res scrolling, large playfield, disk-based Hi-Score list, stereo sound, and multiple levels. Use a joystick in port 2 to control the ship. B, shareware ($8). By: Oliver Wagner

Fme Nicely done map editor for the Fire-Power (tm) game. Features interlaced hi-res with intuition interface. See the "Readme.fnl" file for information on making a bootable disk. Includes source. Author: Gregory MacKay

HandyIcons Adds a menustrip to the WorkBench window that allows you to run selected Workbench Tools by menu selection. Can be set up to provide custom environments. Current version supports only WorkBench Tools and not Projects. Binary by: Alan Rubright

Scrambler A simple program that will encode/decode a text file into illegible gibberish, which resembles executable code, to evade prying eyes. Version 0.01, Binary only. Author: Foster Hall

Fred Fish Disk 149

AnimalSounds A sample of digitized animal sounds along with a simple sound player. Authors: The Trumor Company, Inc. Sound Player by Don Pitts

DX-VoiceSorter Written to be used with Jack Deckard's VoiceFiller program. (Disk 82). It allows for the sorting of a number of voicefiles stored using that program into a new voicefile of voices made up from various files. Includes source. Author: David Bouckley

Keep A nice little utility program with an intuition interface for BBS and network junkies who download messages in one large file and then read them off-line. Using only the mouse, you can drive through such files a message at a time, examine each at your leisure and tag those you wish to keep. Version 1.2, binary only, but source available with donation to author. Author: Tim Grantham

Less Like Unix "more", only better, with forward and backward scrolling, searching and positioning by percent of file and line number, etc. Now lets you also print the current file. Very useful! This is Amiga version 1.3, an update to the version on disk number 92. Includes source. Author: Mark Nudelman, Amiga port by Bob Leivian

Scheme "Scheme is a statically scoped and properly tail-recursive dialect of the Lisp programming language invented by Guy Lewis Steele Jr. and Gerald Jay Sussman." Binary only. Amiga port by Ed Puckett

Fred Fish Disk 150

AirFoil An update to the Airfoil generator on disk #71. Generates airfoil models as well as their corresponding streamline and pressure distributions. Includes source. Authors: Russell Leighton Addendum by David Foster

DC10 An AmigaBasic DC-10 instrument flight simulator. Appears to be quite in-depth with flight-planning and take-off options along with an extensive documentation file. Requires rebuilding on a separate disk and was successfully done so by following the author's instructions in the ReadMe_First file. Author: Jan Arkesteijn

ExecLib A working example of how to build and use user-defined disk-resident libraries. Of special interest to developers working with Lattice C. Author: Alex Livshits

Iconizer A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only. Author: Alex Livshits

Pilot An implementation of the PILOT language for the Amiga, including a demo done for the National Park Service. PILOT is a limited use language for use in educational and computer based instruction programs. Binary only with Beta test kit available from authors. By: T. LaGrone

StealMemBoot A small utility designed to be a direct replacement for NoFastMem kind of programs. It modifies the boot block of a disk, so when you boot with it, all memory allocations will return only CHIP memory. Author: Alex Livshits

Fred Fish Disk 151

GlobeDemo Graphics demo displays very smooth transitions of the rotating earth. Pop-up menu. Source. by: Bob Corwin

Icons Yet another potpourri of interesting icons to choose from if you need one for your own program. by: Dave Turnock

Pcopy A small intuition-based disk copier similiar to the resident "DiskCopy" except with write-verify and other user-selectable options. Useful for making multiple copies with reliable data. Requires two disk drives. S. by: Dirk Reisig

SCT A CLI-based utility (SetCplorTable) for displaying and/or setting a screen's colors. Save the colors of a screen to be restored later, or copy one screen's colors to another. Includes source. Author: aklevin

SlideShow Very nicely done slide-show program written in assembly language. Features forward/backward presentation and creative screen wipes. Currently works only with IFF lo-res pictures. Executable only along with some new IFF pictures to have come my way. Shareware ($16). Authors: Mike McKittrick and Sheldon Templeton

Surveyor A little utility that opens a window on the current screen and displays information about the pointer. Allows for absolute or relative measurement between two points on the screen. Very handy for precise positioning of icons and such. Includes source. Author: Dirk Reisig

Fred Fish Disk 152

Blk A requester making tool employing various recursive algorithms including a recursive parser. It takes input text files and converts them to C-source for including as requester declarations. Includes source. Author: Stuart Ferguson

RunBack A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By Daniel Barrett

UUCP This is a version of uucp (Unix to Unix Copy Program) for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Author: Various, submitted by William Loftus

Fred Fish Disk 153

Dme Version 1.30 of Matt's text editor. Dme is a simple WYSIWYG editor designed for programmers. It is not a WYSIWYG word processor in the traditional sense. Features include arbitrary key mapping, fast scrolling, title-line statistics multiple windows, and ability to iconify windows. Update to version on disk number 134, includes source. by: Matt Dillon

HP11 Emulates an HP11C calculator including the program mode. Features an ON/OFF button that turns the calculator into an icon that will sit and wait until you need it again. Documentation on the features is scarce, perhaps some industrious HP owner could write a small tutorial for the benefit of those that don't own an HP calculator. Binary only. Author: David Gay

HPMam A program to manipulate settings and fonts on HP LaserJet+ printers and compatibles. Includes an intuition interface and some sample picture files. Version 1.0, binary only, shareware. by: Steve Robb

Synthemania An interesting, very small (and very persistent!) musical piece. If you plan on stopping it without using three fingers, you better read the document file first! Binary only. by: Holger Lubitz

Fred Fish Disk 154

Ada An Ada Syntax checker for the amiga. Includes lex and yacc source. Author: Herman Fischer; updates by William Loftus

AssemblyDemos A interesting group of assembly language demos for your visual and aural pleasure. Binary only. by: Foster Hall

DiskLib Two utilities for those people who like to split up PD disks into disks of different categories. Includes source. by: Wilson Snyder

Guardian Another virus diagnosing and vaccination program. Recognizes any non-standard bootblock. Includes a small utility program to permanently place the program on a copy of your kickstart disk in place of the seldom (if ever!) used Debug() function. Binary only. by: Leonardo Fei

PrintSpool A print-spooling program. Very useful for printing files in the background. Many command-line options. Version 1.0.0, Includes source. Author: Frangois Gagnon

Utilities A group of four little utility programs:

UnDelete - Undelete a file from floppy (DF0:) to any device you request, checks for a disk in the drive and allows you to abort cleanly with a CTRL 'C'.

WhereIs - Looks for a file and or directory defaults to the current device

CAL - Clone of the Unix CAL , dates fromyear 1 to 9999.

DClock Simple title bar clock/memory gauge with pop to front.

VirusX An update to the virus-detecting program of the same name on disk number 137. This version also checks for the Byte-Bandit strain. Version 1.21, Includes source. by: Steve Tibbett

Virus_Alert! Yet another anti-virus program with a twist. Once installed a message is displayed just after a warm or cold boot notifying the the user that the disk and memory are virus-free, and forcing a mouse-button press before continuing. Anything writing to the bootblock thereafter will destroy the message and a normal virus- infected boot (???) will take-place. Versions 1.01 and 2.01, Binary only. by: Foster Hall

Wicon A "Window iconifier". Allows you to turn your windows into small icons which can be later recalled. Currently installed with MacWin to give your windows a "rubber-banding" effect. Version 1.14, Includes source. by: Steven Sweeting Introducing the Amiga

Fred Fish Disk 155

AsmExamples A couple of assembly code examples. Executables are not extremely useful but the code might be of considerable benefit to beginning assembly languageprogrammers. by: Henrik Clausen

Bison A replacement for unix "yacc" command. From the GNU (GNU is Not Unix) effort. Contains updates to the version on disk number 136, submitted by two separate sources. Includes source. by: Bob Corbett and Richard Stallman, updates by William Loftus & Scott Henry

NoSmoking Sample program showing the use of a recoverable alert while displaying a personal health message. Includes source. Author: Theo Kermanidis

Scenery A very nice assembly language random scenery generator. Generates very realistic looking landscapes. Includes intuition interface and lots of menu options. V. 1.0, binary only by; Brett Casebolt

Fred Fish Disk 156

Blocks2 Amusing and colorful display of a moving trail of"blocks". Update to version on disk number 71, however this version also Source. by: Gary Walker

Flex Flex is a replacement for the UNIX "lex" (lexical analyzer generator) program that is faster than lex, and freelyredistributable. Includes source. Authors: Jef Poskanzer, Vern Paxson, et. al. Submissions by William Loftus and Scott Henry

Go64 Another screen hack aimed at an earlier Commodore product (Not to be confused with the commercial product Go-64! fromSoftware Insight Systems). Includes source. by: Joerg Anslik

Grammars A group of lexical grammar files for Ada, C and Pascal for use in conjunction with the flex program on this disk and thebison program on disk #155 by: Various, submitted byWilliam Loftus

OOPS! Tired of the monochrome background color of your Workbench or CLI? Then try this colorful screen hack to brighten things upIncludes source. by Joerg Anslik

Fred Fish Disk 157

60or80 A small utility to toggle the 60/80 column text modes without having to go through preferences. Works from either the CLI or the Workbench. S by: Mark Schreten

AmicForm Creates a phonebook containing only those areacodes and exchanges reachable through PC-Pursuit. Input any of Chet Solace's Finalist BBS lists and it creates the phonebook in a form usable byAmicTerm other popular terminal programs.V. 1.3, Binary only. by: J. Motsinger

AnimBalls A nifty little animation program that allows you to create a collection of balls in three-space and then interactively rotate them in real time using the mouse. Includes source.by: Jim Guilford

For PDS orders, please use form on page 112

BootBack A handy little utility to copy and save the boot block from a disk, then later restore it should the disk get stomped on by some ugly virus. Source. by: David Joiner

ECPM A CP/M emulator for the Amiga.Emulates an 8080 along with H19 terminal emulation. Update from version on disk number 109.Source.by: Jim Cathey; port by Charlie Gibbs; Significant improvements by Willi Kusche

KeyFiler BBS message file sorter that allows sorting by keyword. Includes a textreader, Soundex matching, and limited wildcardcapabilities. V. 1.0, Binary only.by: John Motsinger

ScreenZap A little utility to clean away screens that are left by illbehaving programs. It will kill every screen behind the WorkBench, noting how many it gets. The screens in front of WB are not affected. Includes source. Author: Lars Clausen

SetPrefs Allows you to build a whole library of preference settings and instantly switch back and forth between them. Affects allpreference settings not just the colors. Very useful for machines with multiple users or multiple external devices. Includes Amiga's default and various sample preference settings. Binary only. Author: Martin Hippele

Xicon Xicon lets you use icons to call up scripts containing CLI commands. This is version 2.01, an update to the version on disk 102. Includes source. Author: Pete Goodeve

Fred Fish Disk 158

DiskX Nicely done Sector-based disk editor. Binary only by: Steve Tibbett

MemBoardTest Originally designed for production testing of A1000 memory boards. Very nice intuition interface. Version 2.4, Source in Modula by: George Vokalek

MSDOS A program to list files written in standard MS-Dos orAtari ST format. The files can then be copied to Ram and rewritten to disk in Amiga-Dos format. Binary only, Shareware, V. 0.1.Author: Frank Wjbeling

PCBTool An early version of a shareware PC Board layout program Lots of options including variable size pads and traces, grids, grid snap, layers, zoom, selectable centering, text and more. This version does not support printer/ plotter dumps or libraries. V.2.6, binary only.Author: George Vokalek

ScreenX A handy little background utility that provides a small clock/memory counter in its inactive mode and a versatile screen manipulator when called upon. Binary only with source available from author, Version 2.1.Author: Steve Tibbett

TaskX A "real-time" task editor. Lets you list and set the priorities of all the currently running tasks. Binary only, V. 2.0. Author: Steve Tibbett

VirusX Update to the version on disk number 154, checks for a couple of additional new strains. Includes source V. 1.6.Author: Steve Tibbett

YachtC3 Update to the Yachtc program on disk #10, contains some fixes and incorporates a simple sound process. Version 3, includes source. Author: Sheldon Leemon, with enhancements by MarkSchretlen

Fred Fish Disk 159

Free A little command to put in your c directory that returns memory status and number of tasks currently served by EXEC Includes source. Author: Joerg Anslik

MidiTools A group of several different utility programs for those who run a Midi system. Binary only. Author: Jack Deckard

StarChart Nicely done intuition based program to display and identify about 600 stars, galaxies and nebulae visible in the Northern hemisphere. V.1.2, includes source. Author: Ray R. Larson

TaskControl Nicely done task-handling program allowing you to put to sleep, kill or change priorities of the all the currently loaded tasks.Also potentially GURU-producing, so be careful what tasks you kill, change priorities of, etc. Handy windowizer will reduce it almostto an icon Binary only.by: J. Martin Hippele

TUC "The Ultimate Clock". Another window title clock/ memory minder. This one is in 132 columns! Also gives the free memory on drives DF0, DF1 & DF2.Includes source. Author:Joerg Anslik

Fred Fish Disk 160

Calls A little utility to help analyze the flow of a C-program by laying out the functions called in a hierarchical manner.Author: Originally from Usenet with major revisions by Kevin Braunsdorf, Amiga portby George MacDonald

Check A useful little utility for finding structural errors in C-source code. Many command-line options. V.1.03, binary only.Author: Keith Elbertson

Dis A 68000 disassembler, written in assembly, this is an update to the version on disk #128. Includes source Author: Greg Lee with enhancements by Willi Kusche

DMouse A versatile screen & mouse blanker, auto window activator, mouse accelerator, popcli, pop window to front, push window to back, etc, widget. V. 1.09, includes source Update to FF 145 by: Matt Dillon

DWIP "Daisy Wheel IFF Printer". A graphics printing utility that allows the printing of IFF pictures on a daisy wheel printer. Includes source. Author: Ken Van Camp

M4 A UNIX M4 look-alike macro processor intended as a front end Ratfor, Pascal, and other languages that do not have a built-in macro processing capability. Pd M4 reads standard input, the processed text is written on the standard output. Author: Ozan S. Yigit (oz)

MemoPad A shareware intuition-based memo reminder program.Nicely done. Update to version on disk #146, v .2, binary only.Author: Michael Griebling

NeuralNets A neural network example using the general-ized back-propagation delta rule for learning, specifically applied to the tabularasa Little Red Riding Hood instance. by: J. C. Hoskins

Fred Fish Disk 161

Friends Sscreen hack with command-line options to keep your mouse pointer company when you step away. Source by: Michael Warner

Getsprite Simple little program to convert Dpaint brushes into C-sourceBinary only. by: Michael Warner

IncRev A handy little program that will automatically increment the revision number of a program every time it is recompiled.Binary only. by: Bryan Ford

LGZ A Map generator/editor for the LGZ game. Not useful if you don't happen to play that game, but good source example of intuition interfacing. V.0.1 by: Lars & Henrik Clausen

Mackie A versatile cli/macro-key initiator based on POPCLI with a unique method of "screen-blanking". I won't say more, just try it!V.1.1, includes source. Author: Thomas Rokicki

Nag A shareware appointment calendar with it's own editor and a unique 'nagging' feature utilizing the Amiga's voice and audio devices. V.1.6, binary only. by: Richard Lee Stockton

Perl Practical Extraction and Report Language, an interpreted language optimized for scanning arbitrary text files, extracting information from those text files, and printing reports based on that information. by: Larry Wall

VRTest Another anti-virus utility that allows visual inspection of ram starting a $7E7FE, ram cleaning, bootblock inspection and vector monitoring/ reseting. Written in assembly language V. 3.2, binary only.by: Babar Khan

XBoot A very simple utility to convert a boot block into an executable file so you can use your favorite debugger (Wack, Dis, etc.) to study it. Includes source.Author: Francois Rouaix

Fred Fish Disk 162

Avi A workalike version of the UNIX vi editor for the amiga.Thoug not especially recommended for beginners, designed for those of you who may have the vi commands permanently hard-coded into your fingertips! V.1.0, binary only. Author: Peter Nestor

CLI_Utilities This directory contains several subdirectories with small utilities, collected from various sources, that are only usable from the CLI. Some include source. by: Various

Dark A small graphics and animation demo.Includes source. Author: Phil Robertson

Flow2Troff A little utility to convert from New Horizons Software "FLOW" files to UNIX "troff" files, suitable for printing on any troff-compatible laser printer. V. 1.0, includes source and a sample "FLOW" file. Author: Daniel Barrett

LabyrinthII A shareware role-playing text adventure game similar in operation to the Infocom text adventures.Includes source. Author: Russell Wallace

Iffar Maintains archives of Interchange File Format (IFF) FORM CAT and LIST files in a manner that complies with the IFF CAT specification. V.1.2, includes source Author: Karl Lehenbauer

SetPALorNTSC A couple of utility programs for testing the suitability of a developed program in either the PAL or NTSC environments. Includes source and a sample program.Author: Peter Kittal

TES "The Electronic Slave" adds a gadget strip to the top of he cli window to perform such functions as device directories info, run ED, and time. Currently, assignments are hardcoded but not difficult to change if you own a compiler. V. 1.1, includes source.Author: Joerg Anslik

UnknownGirl Another small musical piece similar in execution to Synthemania" on disk number 153. Binary only. AuthorHolger Lubitz

To Be Continued.....

In Conclusion

To the best of our knowledge, the materials in this library are freely distributable. This means they were either publicly posted and placed in the public domain by their authors, or they have restrictions published in their files to which we have adhered. If you become aware of any violation of the authors' wishes, please contact us by mail.

IMPORTANT NOTICE!

This list is compiled and published as a service to the Commodore Amiga community for informational purposes only. Its use is restricted to non-commercial groups only! Any duplication for commercial purposes is strictly forbidden. As a part of Amazing Computing™, this list is inherently copyrighted. Any infringement on this proprietary copyright without expressed written permission of the publishers will incur the full force of legal actions.

Any non-commercial Amiga user group wishing to duplicate this list should contact:

PiM Publications, Inc.
P.O.Box 869
Fall River, MA 02722

PiM Publications Inc. is extremely interested in helping any Amiga user groups in non-commercial support for the Amiga.

•AC•

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

1A. Title of Publication: Amazing Computing. 1B. Publication No.: 0886-9480. 2. Date of Filing: 10/ 5/88. 3. Frequency of Issue: Monthly. 3A. No. of Issues Published Annually: 12. 3B. Annual Subscription Price: $24. 4. Complete Mailing Address of Known Office of Publication: One Currant Place, Currant Rd., P.O. Box 869, Fall River, MA 02722-0869. 5. Complete Mailing Address of the Headquarters of Genral Business Offices of the Publisher: One Currant Place, Currant Rd., P.O. Box 869, Fall River, MA 02722-0869. 6. Full Name and Complete Mailing Address of Publisher, Editor, and Managing Editor: Publisher, Joyce A. Hicks, P.O. Box 869, Fall River, MA 02722-0869; Editor, Ernest P. Viveiros, Jr, P.O. Box 869, Fall River, MA 02722-0869; Managing Editor, Donald D. Hicks, P.O. Box 869, Fall River, MA 02722-0869. 7. Owner: PiM Publications, Inc., P.O. Box 869, Fall River, MA 02722-0869; Joyce A. Hicks, P.O. Box 869, Fall River, MA 02722-0869; Donald D. HIcks, P.O. Box 869, Fall River, MA 02722-0869. 8. Known Bondholders, Mortgagees and Other Security Holders Owning or Holding 1 Percent or More of Total Amount of Bonds, Mortgages or Other Securities: None. 9. For Completion by Nonprofit Organizations Authorized to Mail at Special Rates: Not Applicable. 10. Extent and Nature of Circulation: (X) Average No. Copies Each Issue During Preceding 12 Months (since United States second class postage was only in effect on this publication for five months prior to this filling, to establish an average, the enclosed numbers can only and do only refer to the last five months); (Y) Actual No. Copies of Single Issue Published Nearest to Filing Date; 10A. Total No. Copies (X) 43,991 (Y) 45,191. 10B. Paid Circulation: 1. Sales through dealers and carriers, street vendors and counter sales: (X) 23,058 (Y) 27,504. 2. Mail Subscriptions: (X) 10,331 (Y) 10,633. 10C. Total paid and/or Requested Circulation: (X) 33,389 (Y) 38,137. 10D. Free Distribution by Mail Carrier or Other Means Samples, Complimentary, and Other Free Copies; (X) 353 (Y) 521. 10E. Total Distribution: (X) 33,742 (Y) 38,658. 10F. Copies Not Distributed: 1. Office Use, left over*, unaccounted, spoiled after printing: (X) 6,607 (Y) 6,134. 2. Return from News Agents: (X) 3,642 (Y) 399. 10G. Total: (X) 43,991 (Y) 45,191.

*Issues held for back issue sales.

Amaze Me

Please use this order form when subscribing to Amazing Computing™, ordering Back Issues, or ordering Amiga™ Public Domain Software

Name_____
Street_____
City_____St._____Zip_____
Amount Enclosed_____

Please circle the appropriate item: New Subscription Renewal

Subscription: $_____

Please start my subscription to Amazing Computing™
with the next available issue or renew current subscription.
I have enclosed $24.00 for 12 issues in the U.S.
($36.00 Canada and Mexico, $44.00 foreign surface).
All funds must be in U.S. Currency on a U.S. Bank

Back Issues:
$5.00 each US, $6.00 each Canada and Mexico, $7.00 each Foreign Surface.

Please circle your Back Issue choices below:

Vol1.1	Vol1.2	Vol1.3	Vol1.4	Vol1.5	Vol1.6	Vol1.7	Vol1.8	Vol1.9	Vol2.1	Vol2.2
Vol2.3	Vol2.4	Vol2.5	Vol2.6	Vol2.7	Vol2.8	Vol2.9	Vol2.10	Vol2.11	Vol2.12	Vol3.1
Vol3.2	Vol3.3	Vol3.4	Vol3.5	Vol3.6	Vol3.7	Vol3.8	Vol3.9	Vol3.10	Vol3.11	

Back Issues: $_____

Public Domain Software:
$6.00 each for subscribers (yes, even the new ones!)
$7.00 each for non subscribers
(three disk minimum on all foreign orders)

Please circle your Public Domain Software choices below:

Amazing on Disk:

A#1. . .Source & Listings V3.9

Amicus:

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17
A18	A19	A20	A21	A22	A23	A24	A25	A26								

Fred Fish:

FF1	FF2	FF3	FF4	FF5	FF6	FF7	FF8	FF9	FF10	FF11	FF12	FF13	FF14	FF15	FF16	FF17
FF18	FF19	FF20	FF21	FF22	FF23	FF24	FF25	FF26	FF27	FF28	FF29	FF30	FF31	FF32	FF33	FF34
FF35	FF36	FF37	FF38	FF39	FF40	FF41	FF42	FF43	FF44	FF45	FF46	FF47	FF48	FF49	FF50	FF51
FF52	FF53	FF54	FF55	FF56	FFNA	FF58	FF59	FF60	FF61	FF62	FF63	FF64	FF65	FF66	FF67	FF68
FF69	FF70	FF71	FF72	FF73	FF74	FF75	FF76	FF77	FF78	FF79	FFNA	FF81	FF82	FF83	FF84	FF85
FF86	FF87	FFNA	FF89	FF90	FF91	FF92	FF93	FF94	FF95	FF96	FF97	FF98	FF99	FF100	FF101	FF102
FF103	FF104	FF105	FF106	FF107	FF108	FF109	FF110	FF111	FF112	FF113	FF114	FF115	FF116	FF117	FF118	FF119
FF120	FF121	FF122	FF123	FF124	FF125	FF126	FF127	FF128	FF129	FF130	FF131	FF132	FF133	FF134	FF135	FF136
FF137	FF138	FF139	FF140	FF141	FF142	FF143	FF144	FF145	FF146	FF147	FF148	FF149	FF150	FF151	FF152	FF153
FF154	FF155	FF156	FF157	FF158	FF159	FF160	FF161	FF162								

PDS Disks: $_____

(NA Denotes disks removed from the collection)

Please complete this form and mail with check or money order to:

PiM Publications, Inc.
P.O. Box 869
Fall River, MA 02722-0869

Total: $_____

Please allow 4 to 6 weeks for delivery

Amazing Computing Reader Service Card

Want to know more about Amazing computing advertisers and their products? Here's yourchance! Fill out the card below (for instructions,see page 96) and drop it in the mail (U.S. only). We will process your request promptly and forward it to the advertiser.

Quick easy, and efficient—just like all services of Amazing Computing: Your Original Amiga Monthly Resource.

Amazing Computing Reader Service Card

Name_____

Street_____

City _____ St.____ Zip _____

Telephone_____

December card Valid until 2/28/89

Please see page 96 for instructions.

A. Do you own an Amiga?
- O 1. Amiga 1000
- O 2. Amiga 500
- O 3. Amiga 2000
- O 4. Soon
- O 5. Not Yet
- O 6. Just Looking

B. What Amiga hardware product do you plan to buy next?
- O 1. Amiga 500
- O 2. Amiga 2000
- O 3. Memory Expansion
- O 4. Hard Drive
- O 5. IBM Emulators (sidecar or Bridgeboard)
- O 6. Printer
- O 7. Modem
- O 8. Music Tool
- O 9. Video Product
- O 10. Other

C. What Amiga software product do you plan to buy next?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Entertainment
- O 7. Telecommunications
- O 8. Spreadsheet
- O 9. Database
- O 10. Financial
- O 11. Video
- O 12. Graphics
- O 13. Music
- O 14. Other

D. Where do you buy your Amiga products?
- O 1. Local Amiga Dealer
- O 2. Discount Department Store
- O 3. Manufacturer
- O 4. Mail Order

E. Which type of articles do you like to see in Amazing Computing?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Game Reviews
- O 7. Business Reviews
- O 8. Hardware Product Reviews
- O 9. Software Product Reviews
- O 9. Programming How To's
- O 10. Business How To's
- O 11. Video Articles
- O 12. Graphics Articles
- O 13. Music Articles
- O 14. Hardware How To's
- O 15. PDS Updates
- O 16. Interviews
- O 17. Other

F. Which articles would you like to see more of in Amazing Computing?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Game Reviews
- O 7. Business Reviews
- O 8. Hardware Product Reviews
- O 9. Software Product Reviews
- O 9. Programming How To's
- O 10. Business How To's
- O 11. Video Articles
- O 12. Graphics Articles
- O 13. Music Articles
- O 14. Hardware How To's
- O 15. PDS Updates
- O 16. Interviews
- O 17. Other

G Are you a subscriber to Amazing Computing?
- O 1. Yes
- O 2. No

101

101	102	103	104	105	201	202	203	204	205
106	107	108	109	110	206	207	208	209	210
111	112	113	114	115	211	212	213	214	215
116	117	118	119	120	216	217	218	219	220
121	122	123	124	125	221	222	223	224	225
126	127	128	129	130	226	227	228	229	230
131	132	133	134	135	231	232	233	234	235
136	137	138	139	140	236	237	238	239	240
141	142	143	144	145	241	242	243	244	245
146	147	148	149	150	246	247	248	249	250
151	152	153	154	155	251	252	253	254	255
156	157	158	159	160	256	257	258	259	260
161	162	163	164	165	261	262	263	264	265
166	167	168	169	170	266	267	268	269	270
171	172	173	174	175	271	272	273	274	275
176	177	178	179	180	276	277	278	279	280
181	182	183	184	185	281	282	283	284	285
186	187	188	189	190	286	287	288	289	290
191	192	193	194	195	291	292	293	294	295
196	197	198	199	200	296	297	298	299	300

Amazing Computing Reader Service Card

Name_____

Street_____

City _____ St.____ Zip _____

Telephone_____

December card Valid until 2/28/89

Please see page 96 for instructions.

A. Do you own an Amiga?
- O 1. Amiga 1000
- O 2. Amiga 500
- O 3. Amiga 2000
- O 4. Soon
- O 5. Not Yet
- O 6. Just Looking

B. What Amiga hardware product do you plan to buy next?
- O 1. Amiga 500
- O 2. Amiga 2000
- O 3. Memory Expansion
- O 4. Hard Drive
- O 5. IBM Emulators (sidecar or Bridgeboard)
- O 6. Printer
- O 7. Modem
- O 8. Music Tool
- O 9. Video Product
- O 10. Other

C. What Amiga software product do you plan to buy next?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Entertainment
- O 7. Telecommunications
- O 8. Spreadsheet
- O 9. Database
- O 10. Financial
- O 11. Video
- O 12. Graphics
- O 13. Music
- O 14. Other

D. Where do you buy your Amiga products?
- O 1. Local Amiga Dealer
- O 2. Discount Department Store
- O 3. Manufacturer
- O 4. Mail Order

E. Which type of articles do you like to see in Amazing Computing?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Game Reviews
- O 7. Business Reviews
- O 8. Hardware Product Reviews
- O 9. Software Product Reviews
- O 9. Programming How To's
- O 10. Business How To's
- O 11. Video Articles
- O 12. Graphics Articles
- O 13. Music Articles
- O 14. Hardware How To's
- O 15. PDS Updates
- O 16. Interviews
- O 17. Other

F. Which articles would you like to see more of in Amazing Computing?
- O 1. 'C' Language
- O 2. Forth Language
- O 3. Modula-2 Language
- O 4. Assembly Language
- O 5. BASIC Language
- O 6. Game Reviews
- O 7. Business Reviews
- O 8. Hardware Product Reviews
- O 9. Software Product Reviews
- O 9. Programming How To's
- O 10. Business How To's
- O 11. Video Articles
- O 12. Graphics Articles
- O 13. Music Articles
- O 14. Hardware How To's
- O 15. PDS Updates
- O 16. Interviews
- O 17. Other

G Are you a subscriber to Amazing Computing?
- O 1. Yes
- O 2. No

102

101	102	103	104	105	201	202	203	204	205
106	107	108	109	110	206	207	208	209	210
111	112	113	114	115	211	212	213	214	215
116	117	118	119	120	216	217	218	219	220
121	122	123	124	125	221	222	223	224	225
126	127	128	129	130	226	227	228	229	230
131	132	133	134	135	231	232	233	234	235
136	137	138	139	140	236	237	238	239	240
141	142	143	144	145	241	242	243	244	245
146	147	148	149	150	246	247	248	249	250
151	152	153	154	155	251	252	253	254	255
156	157	158	159	160	256	257	258	259	260
161	162	163	164	165	261	262	263	264	265
166	167	168	169	170	266	267	268	269	270
171	172	173	174	175	271	272	273	274	275
176	177	178	179	180	276	277	278	279	280
181	182	183	184	185	281	282	283	284	285
186	187	188	189	190	286	287	288	289	290
191	192	193	194	195	291	292	293	294	295
196	197	198	199	200	296	297	298	299	300

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 36 FALL RIVER, MA

POSTAGE WILL BE PAID BY ADDRESSEE

Amazing COMPUTING
P.O. Box 869
Fall River, MA 02722-0869

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 36 FALL RIVER, MA

POSTAGE WILL BE PAID BY ADDRESSEE

Amazing COMPUTING
P.O. Box 869
Fall River, MA 02722-0869

Do It NOW!!!

Subscribe TODAY and save over 49% on Amazing Computing!

Amazing Computing is available to United States subscribers at $24.00 for 12 issues. That's a savings of $23.40 off the newsstand price. Canada and Mexico, $36.00. Foreign Surface to all other countries, $44.00
(Air mail and first class rates available upon request.)

Enter my 12 issue subscription today!

Name_____

Street_____

City _____ St.____ Zip _____

Amount Enclosed _____

Sorry, we cannot accept credit cards or requests for billing at this time.
All funds must be in U.S. Currency drawn on a U.S. Bank.
Please allow four to six weeks for processing.

Clip and $AVE !!

(over 49% of the newsstand price!)

Save Time and Money!

Subscribe TODAY and save over 49% on Amazing Computing!

Amazing Computing is available to United States subscribers at $24.00 for 12 issues. That's a savings of $23.40 off the newsstand price. Canada and Mexico, $36.00. Foreign Surface to all other countries, $44.00.
(Air mail and first class rates available upon request.)

Enter my 12 issue subscription today!

Name_____

Street_____

City _____ St.____ Zip _____

Amount Enclosed _____

Sorry, we cannot accept credit cards or requests for billing at this time.
All funds must be in U.S. Currency drawn on a U.S. Bank.
Please allow four to six weeks for processing.

Please Mail to:

Amazing COMPUTING

P.O. Box 869
Fall River, MA 02722-0869

Place this order form in an envelope with your Check or Money Order.

Clip and $AVE !!

(over 49% of the newsstand price!)

Please Mail to:

Amazing COMPUTING

P.O. Box 869
Fall River, MA 02722-0869

Place this order form in an envelope with your Check or Money Order.

VOTE

Your Civic AMIGA Duty Calls!

What are the best software packages for the Amiga?

Don't ask us.

It's all up to you.

At AmiExpo/New York in March, **YOU** will present awards to the Amiga's top software developers. Sure, we could pick out our favorites, but no one is more qualified than you, the users. You use Amiga software everyday to solve problems, explore creativity, and just have fun. *Amazing* 's job is just to report what you are doing with the Amiga!

And the best part is you don't have to break through crowds of hand-shaking poll hawkers. You don't even have to be at AmiExpo! Just complete the ballot at the bottom of this page and mail it to *Amazing* by February 15, 1989. *Amazing Computing* will tally up your choices and hand out awards directly from your votes.

At AmiExpo/Los Angeles last October, *Amazing Computing* announced awards to individuals who have dedicated their hard work and creative power to the Amiga community. Now we leave it up to you, active members of that same Amiga community, to choose the award-winning products. *Amazing* recognizes individuals, but only you can recognize the most useful products.

What are those priceless programs that bring the power of your Amiga to life and make life in general a whole lot easier? Now rave about your favorites. Stand up and be counted. Cast your Amiga ballot.

STAND UP AND BE COUNTED!

Which Amiga Products Do You Think Are Award-Winners?

Cast your Amiga Ballot now by filling in your choice for the following questions.

Name _____

Address _____

City _____

State _____ Zip _____

SOFTWARE

Choose your preferred:

WORD PROCESSOR: _____

DATABASE: _____

SPREADSHEET: _____

DESKTOP PUBLISHING package: _____

GRAPHICS/PAINT package: _____

CAD/DRAW package: _____

ACCOUNTING package: _____

TELECOMMUNICATIONS software: _____

_____

PROGRAMMING LANGUAGE: _____

PROGRAMMING UTILITY: _____

FILE UTILITY: _____

ANIMATION/VIDEO software: _____

GAME: _____

AUDIO/MUSIC software: _____

EDUCATIONAL software: _____

BUSINESS software: _____

HARDWARE

Choose your preferred:

MASS STORAGE device: _____

ACCELERATOR: _____

DIGITIZER/FRAME GRABBER: _____

PRINTER: _____

MONITOR: _____

INPUT device: _____

MODEM: _____

Overall, the **MOST USEFUL** Amiga product in my library is: _____

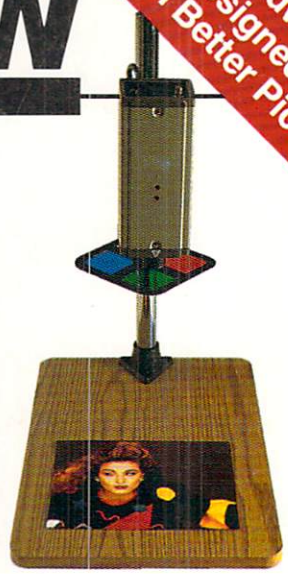
Please return to:
Reader Ballots Amazing Computing
P.O. Box 869 • Fall River, MA 02722

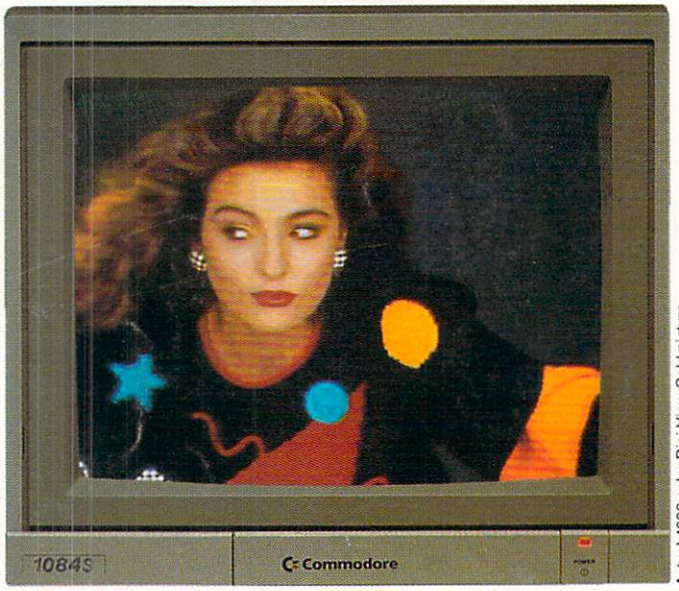
DIGI-VIEW GOLD

All-New!
Hardware and Software Designed for A500/2000—
Even Better Pictures Than Before!

1.

2.





Actual 4096 color Digi-View Gold picture

3. Simply the Best.

The all new Digi-View Gold is the best video digitizer for the Amiga. Period. Nothing else even comes close. Why? The secret is that Digi-View Gold captures 2.1 million colors in memory, giving you an incredible 100,000 apparent colors on screen simultaneously.

And it's easy to use. Just focus your video camera on any object or picture, and in seconds Digi-View Gold turns it into Amiga graphics that glow with vibrant color and clarity. Whether you are creating graphics for desktop publishing, presentations, video, or just for fun, Digi-View Gold gives you dazzling images with amazing simplicity.

Digi-View Gold is designed specifically for the Amiga 500 and 2000, and plugs directly into the parallel port. Digi-View Gold's powerful image capture and manipulation software (version 3.0) now has complete control of color and sharpness, full overscan, extra halfbrite, and a special line art mode for desktop publishing.

Only Digi-View Gold:

- Can digitize in all Amiga resolution modes from 320x200 up to 768x480 (full hi-res overscan)
- Uses 2 to 4096 colors (including extra halfbrite)
- Uses exclusive Enhanced HAM for super fine detail
- Is 100% IFF compatible and works with any graphics software
- Can digitize 21 bits per pixel (2.1 million colors) for the highest quality images possible
- Has advanced dithering routines that give an apparent 100,000 colors on screen simultaneously
- Has powerful Image processing controls for complete IFF picture manipulation

If you want the highest quality graphics for your Amiga, as easy as 1, 2, 3; then you need the new version of the best selling video digitizer of all time: Digi-View Gold.

*Requires standard gender changer for use with Amiga 1000. Video camera required; not included. NewTek sells a video camera, copy stand, and the Digi-Droid automated filter wheel for Digi-View Gold. If your local retailer doesn't carry these products, call us at 913-354-1146. Digi-View Gold is a trademark of NewTek, Inc. Amiga is a trademark of Commodore-Amiga, Inc. Be seeing you!!

Only $199.95

Digi-View Gold is available now at your local Amiga dealer.
Or call 1-800-843-8934

NewTek
INCORPORATED